

(43) International Publication Date
22 December 2016 (22.12.2016)

- (51) **International Patent Classification:**
H04W 4/00 (2009.01) *H04L 29/08* (2006.01)
- (21) **International Application Number:**
PCT/US2016/037623
- (22) **International Filing Date:**
15 June 2016 (15.06.2016)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
62/175,548 15 June 2015 (15.06.2015) US
- (71) **Applicant:** INTERDIGITAL TECHNOLOGY CORPORATION [US/US]; 200 Bellevue Parkway, Suite 300, Wilmington, Delaware 19809 (US).
- (72) **Inventors:** LOEB, Shoshana; 1001 E. Hector Street, Suite 300, Conshohocken, Pennsylvania 19428 (US). LEE, Kuo-Chu; 30 Revere Court, Princeton Junction, New Jersey 08550 (US).
- (74) **Agent:** LA BARR, Edward T.; Volpe and Koenig, P.C., 30 S. 17th Street, United Plaza, Philadelphia, Pennsylvania 19103 (US).

- (81) **Designated States** (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) **Designated States** (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

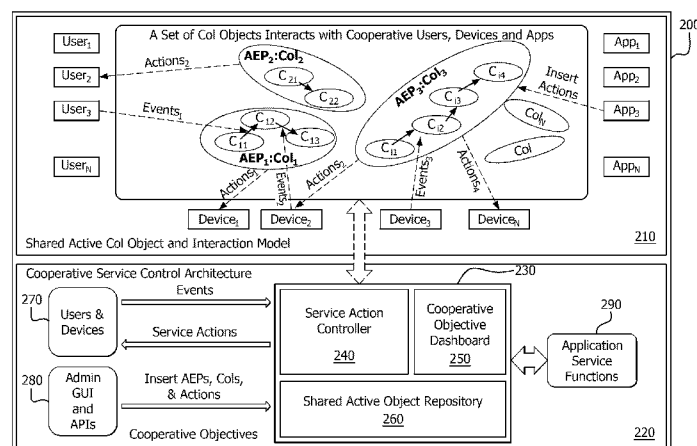
(54) **Title:** METHODS AND SYSTEM FOR CONTROLLING COOPERATIVE CONTEXT AWARE IOT SERVICES

FIG. 2

(57) **Abstract:** Methods and systems to allow multiple independently developed service functions to cooperate and offer service actions to users and devices dynamically, incrementally, and efficiently. The methods and systems may be deployed to distributed internet of things (IoT) environments. Embodiments support dynamic creation of contexts of interests (CoIs) based on combinations of real-time event patterns, user behavior metrics, activities, and environment conditions. The CoI creation, detection and service action control executions are decoupled from the service function creation and execution. Each application and service may insert new service action to the action list to invoke remote execution of service functions dynamically. The decoupled detection, action control and service functions execution, reduce the duplicated design and processing efforts. In addition, coordination of actions at the moments of engagements defined by shared CoIs create opportunities for multiple service providers to offer bundled services cooperatively and incrementally while minimize resource conflicts and service disruption.

METHODS AND SYSTEM FOR CONTROLLING COOPERATIVE CONTEXT AWARE IOT SERVICES

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Provisional Application Serial No. 62/175,548 filed June 15, 2015, the contents of which are hereby incorporated by reference herein.

BACKGROUND

[0002] Advanced Internet of Things (IoT) communications and data analysis are beginning to see applications in areas such as healthcare, energy management, and agriculture. Current IoT platforms may support data collection and control interfaces to multiple distributed IoT devices.

SUMMARY

[0003] Some embodiments provide a method for controlling context aware internet-of-things (IoT) services. Receiving circuitry of a service server receives an actionable event pattern (AEP) and a list of actions corresponding to the AEP. Each action in the list of actions has a priority. The receiving circuitry of the service server receives event information from an event generating device. Detecting circuitry of the service server detects whether the event information satisfies the AEP. If the event information satisfies the AEP, the service selecting circuitry of the service server selects an action from the list of actions based on the priority of the actions in the list of actions and transmitting circuitry of the service server transmits action information based on the selected action, to an actuating device.

[0004] In some embodiments, the receiving circuitry of the service server may receive feedback regarding an action performed by the actuating device based on the action information transmitted to the actuating device and measuring circuitry of the service server may measure an effectiveness of the action. Adjusting circuitry of the service server may adjust the priority of the action based on the measured effectiveness. Adjusting circuitry of the service

server may adjust the priority of at least one action of the list of actions based on the measured effectiveness. The effectiveness may include a cost associated with the action. The feedback may be received by the service server from the actuating device. The event information may include information regarding a user behavior. The event information may include IoT sensor data. Selecting the action from the list of actions based on the priority of the actions in the list of actions may include selecting the highest priority action from the list of actions which does not conflict with another action. The action may not conflict with another action if the action does not reduce an effectiveness of the other action.

[0005] Some embodiments provide a service server for controlling context aware internet-of-things (IoT) services, the service server includes a processor and a non-transitory computer readable memory. The service server also includes receiving circuitry for receiving an actionable event pattern (AEP) and a list of actions corresponding to the AEP, wherein each action in the list of actions has a priority, and for receiving, event information from an event generating device. The service server also includes detecting circuitry for detecting whether the event information satisfies the AEP and selecting circuitry for, if the event information satisfies the AEP, selecting an action from the list of actions based on the priority of the actions in the list of actions. The service server also includes transmitting circuitry for, if the event information satisfies the AEP, transmitting action information based on the selected action, to an actuating device.

[0006] In some embodiments the receiving circuitry of the service server may receive feedback regarding an action performed by the actuating device based on the action information transmitted to the actuating device. The service server may include measuring circuitry for measuring an effectiveness of the action. The service server may include adjusting circuitry of the service server for adjusting the priority of the action based on the measured effectiveness. The service server may include adjusting circuitry for adjusting priority of at least one action of the list of actions based on the measured

effectiveness. The effectiveness may include a cost associated with the action. The feedback may be received by the service server from the actuating device. The event information may include information regarding a user behavior. The event information may include IoT sensor data. Selecting the action from the list of actions based on the priority of the actions in the list of actions may include selecting the action from the list of actions having the highest priority which does not conflict with another action. The action may not conflict with another action if the action does not reduce an effectiveness of the other action.

[0007] Some embodiments provide a method for controlling context aware internet-of-things (IoT) services. Receiving circuitry of a service server receives event information from an event generating device. Updating circuitry of the service server updates context information in a memory based on the event information. Detecting circuitry of the service server detects whether the updated context information corresponds to a context of interest (CoI). If the updated context information corresponds to a CoI the detecting circuitry detects whether the CoI is identified in an actionable event pattern (AEP) function. If the CoI is identified in an AEP function transmitting circuitry of the service server transmits action information to an actuating device.

[0008] In some embodiments, the transmitting circuitry of the service server may transmit a second action information to a second actuating device on a condition that a second AEP identifies the same CoI. The action information may be modified on a condition that a second AEP identifies the same CoI. The transmitting circuitry may refrain from transmitting the action information on a condition that a second AEP identifies the same CoI. The receiving circuitry of the service server may receive configuration information for the CoI from a service provider via a configuration interface. The receiving circuitry may receive configuration information for the AEP from a service provider via a configuration interface. The receiving circuitry may receive configuration information for the CoI from a service provider via a

configuration interface, and second configuration information for a second CoI from a second service provider via the configuration interface. The receiving circuitry may receive configuration information for the AEP from a service provider via a configuration interface, and receive second configuration information for a second AEP from a second service provider via the configuration interface. The event generating device may comprise an IoT sensor. The actuating device may comprise an IoT actuator.

[0009] Some embodiments provide a service server for controlling context aware internet-of-things (IoT) services. The service server includes a processor, a non-transitory computer readable memory, receiving circuitry for receiving event information from an event generating device, updating circuitry for updating context information in the memory based on the event information, detecting circuitry for detecting whether the updated context information corresponds to a context of interest (CoI) and on a condition that the updated context information corresponds to a CoI, detecting whether the CoI is identified in an actionable event pattern (AEP) function, and transmitting circuitry for transmitting action information to an actuating device on a condition that the CoI is identified in an AEP function.

[0010] In some embodiments, the transmitting circuitry may transmit a second action information to a second actuating device on a condition that a second AEP identifies the same CoI. The service server may include modifying circuitry for modifying the action information on a condition that a second AEP identifies the same CoI. The transmitting circuitry may refrain from transmitting the action information on a condition that a second AEP identifies the same CoI. The receiving circuitry may include a configuration interface for receiving configuration information for the CoI from a service provider. The receiving circuitry may include a configuration interface for receiving configuration information for the AEP from a service provider. The receiving circuitry may include a configuration interface for receiving configuration information for the CoI from a service provider via a configuration interface, and for receiving second configuration information for

a second CoI from a second service provider. The receiving circuitry may include a configuration interface for receiving configuration information for the AEP from a service provider via a configuration interface, and for receiving second configuration information for a second AEP from a second service provider. The event generating device may include an IoT sensor. The actuating device may include an IoT actuator.

[0011] In an embodiment, a method of improving effectiveness of service actions offered by multiple service providers to a set of users in a multi-service provider pervasive service environment is disclosed. The method may include: detecting contexts of interest (CoIs) of the users from one or more smart internet of things (IoT) sensors; and providing a shared service interface for the multiple service providers to provision, execute, and prioritize the service actions cooperatively.

[0012] In another embodiment, a method of providing service functions to users at different moments of engagements in an internet of things (IoT) enabled location is disclosed. The method may include: tracking measurable cooperative objectives defined and agreed by participating service functions; defining and detecting evolving contexts of interest (CoIs); collecting events from multiple IoT devices; extracting variables defined in the CoIs; detecting occurrences of event patterns that match the CoIs; developing service functions; deploying the service functions; inserting service entry points into action lists of Actionable Event Patterns (AEPs); creating AEP templates; and executing actions in the action list of triggered AEPs.

[0013] In another embodiment, a method of providing assistance to a shopper is disclosed. The method may include: mapping internet of things (IoT) devices to a virtual 3D environment that supports inventory and location tracking using virtual objects and bots to model shelves, items, shopping carts and shoppers.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] A more detailed understanding may be had from the following

description, given by way of example in conjunction with the accompanying drawings wherein:

[0015] FIG. 1A is a system diagram of an example communications system in which one or more disclosed embodiments may be implemented;

[0016] FIG. 1B is a system diagram of an example wireless transmit/receive unit (WTRU) that may be used within the communications system illustrated in FIG. 1A;

[0017] FIG. 1C is a system diagram of an example radio access network and an example core network that may be used within the communications system illustrated in FIG. 1A;

[0018] FIG. 1D is a system diagram of another example radio access network and an example core network that may be used within the communications system illustrated in FIG. 1A;

[0019] FIG. 1E is a system diagram of an example communications system in which one or more disclosed embodiments may be implemented;

[0020] FIG. 2 is a schematic diagram illustrating example service action control system including an active context of interest (CoI) object model and service architecture;

[0021] FIG. 3 is system diagram illustrating an example internet of things (IoT) enabled pervasive shopping service control system;

[0022] FIG. 4 is a system diagram illustrating an example service control system which includes multiple IoT enabled services;

[0023] FIG. 5 is a flow chart illustrating an example method for detecting and acting on CoIs;

[0024] FIG. 6 is a flow chart illustrating another example method for detecting and acting on CoIs; and

[0025] FIG. 7 is a flow chart illustrating an example method for configuring and employing AEPs.

DETAILED DESCRIPTION

[0026] The contexts in a pervasive Internet of Things (IoT) environment may be complex, application dependent, and constantly evolving. For example, user behavior such as vital signs, orientations, motions, gestures and actions collected from different kind of IoT sensors may change over time according to the evolving user behavior and progression of user activities. Environment conditions such as temperature, humidity, and occupancy may also change and affect the user behavior.

[0027] Context aware complex event pattern detection and service action systems have been proposed for multiple independent application developers and service providers to improve user experiences in real time. In an IoT environment where the sensors and actuators are connected by IoT networking and accessible by multiple services, complex interactions between users and environment under different context may be of interests to more than one high level services. Discussed herein are various techniques to manage the complex, context aware, interaction event patterns and to control one or more service action provisioned by one or more service providers in one or more companies, cooperatively. However, independent service providers may be very hesitant to share customer data with each other, especially context data which could be used to derive a competitive advantage.

[0028] When offering multiple context aware services to the same customers in a shared IoT environment, events from each user may be duplicated and processed in parallel by multiple services subscribed to the same events. For example, after subscribing to IoT brokers, multiple services running in multiple systems (e.g., energy management, fitness, marketing, and customer management systems) may perform complex and resource consuming real-time, location sensitive, behavior tracking, analysis, and event pattern detection functions on the same events from the same set of users and devices in the same environment multiple times independently.

[0029] Furthermore, when multiple service offers are presented to the same users by one or more organizations or service providers independently, out of context or conflicting service actions can occur. This may reduce the effectiveness of services. For example, in a retail store or shopping center, customer support systems may decide when to dispatch service staffs to help customers, marketing system may decide when to send coupons to promote new products, sale automation system may suggest up-sale items, and inventory system may decide when to offer clearance items. When a shopper is looking for an item, a sales person may be dispatched to capture the moment of engagement to help customer and to promote up-sale items at the same time. However, sending coupons and clearance sale items after shoppers have already purchased an up-sale items promoted by sales, may be ineffective and counterproductive.

[0030] Embodiments described herein with reference to FIGS. 1A-6 may include methods and systems to manage the complex, context aware, interaction event patterns and to control one or more service actions provisioned by one or more service providers in one or more companies, cooperatively. Embodiments may be applied to both a cooperative operation across different companies as well as different service providing organizations within one corporation.

[0031] Referring now to FIG. 1A, a diagram of an example communications system 100, in which one or more disclosed embodiments may be implemented, is shown. The communications system 100 may be a multiple access system that provides content, such as voice, data, video, messaging, broadcast, etc., to multiple wireless users. The communications system 100 may enable multiple wireless users to access such content through the sharing of system resources, including wireless bandwidth. For example, the communications systems 100 may employ one or more channel access methods, such as code division multiple access (CDMA), time division multiple access (TDMA), frequency division multiple access (FDMA), orthogonal FDMA (OFDMA), single-carrier FDMA (SC-FDMA), and the like.

[0032] As shown in FIG. 1A, the communications system 100 may include wireless transmit/receive units (WTRUs) 102a, 102b, 102c, 102d, a radio access network (RAN) 104, a core network 106, a public switched telephone network (PSTN) 108, the Internet 110, and other networks 112, though it will be appreciated that the disclosed embodiments contemplate any number of WTRUs, base stations, networks, and/or network elements. Each of the WTRUs 102a, 102b, 102c, 102d may be any type of device configured to operate and/or communicate in a wireless environment. By way of example, the WTRUs 102a, 102b, 102c, 102d may be configured to transmit and/or receive wireless signals and may include user equipment (UE), a mobile station, a fixed or mobile subscriber unit, a pager, a cellular telephone, a personal digital assistant (PDA), a smartphone, a laptop, a netbook, a personal computer, a wireless sensor, consumer electronics, and the like.

[0033] The communications systems 100 may also include a base station 114a and a base station 114b. Each of the base stations 114a, 114b may be any type of device configured to wirelessly interface with at least one of the WTRUs 102a, 102b, 102c, 102d to facilitate access to one or more communication networks, such as the core network 106, the Internet 110, and/or the other networks 112. By way of example, the base stations 114a, 114b may be a base transceiver station (BTS), a Node-B, an eNode B, a Home Node B, a Home eNode B, a site controller, an access point (AP), a wireless router, and the like. While the base stations 114a, 114b are each depicted as a single element, it will be appreciated that the base stations 114a, 114b may include any number of interconnected base stations and/or network elements.

[0034] The base station 114a may be part of the RAN 104, which may also include other base stations and/or network elements (not shown), such as a base station controller (BSC), a radio network controller (RNC), relay nodes, etc. The base station 114a and/or the base station 114b may be configured to transmit and/or receive wireless signals within a particular geographic region, which may be referred to as a cell (not shown). The cell may further be divided into cell sectors. For example, the cell associated with the base station

114a may be divided into three sectors. Thus, in one embodiment, the base station 114a may include three transceivers, i.e., one for each sector of the cell. In another embodiment, the base station 114a may employ multiple-input multiple-output (MIMO) technology and, therefore, may utilize multiple transceivers for each sector of the cell.

[0035] The base stations 114a, 114b may communicate with one or more of the WTRUs 102a, 102b, 102c, 102d over an air interface 116, which may be any suitable wireless communication link (e.g., radio frequency (RF), microwave, infrared (IR), ultraviolet (UV), visible light, etc.). The air interface 116 may be established using any suitable radio access technology (RAT).

[0036] More specifically, as noted above, the communications system 100 may be a multiple access system and may employ one or more channel access schemes, such as CDMA, TDMA, FDMA, OFDMA, SC-FDMA, and the like. For example, the base station 114a in the RAN 104 and the WTRUs 102a, 102b, 102c may implement a radio technology such as Universal Mobile Telecommunications System (UMTS) Terrestrial Radio Access (UTRA), which may establish the air interface 116 using wideband CDMA (WCDMA). WCDMA may include communication protocols such as High-Speed Packet Access (HSPA) and/or Evolved HSPA (HSPA+). HSPA may include High-Speed Downlink Packet Access (HSDPA) and/or High-Speed Uplink Packet Access (HSUPA).

[0037] In another embodiment, the base station 114a and the WTRUs 102a, 102b, 102c may implement a radio technology such as Evolved UMTS Terrestrial Radio Access (E-UTRA), which may establish the air interface 116 using Long Term Evolution (LTE) and/or LTE-Advanced (LTE-A).

[0038] In other embodiments, the base station 114a and the WTRUs 102a, 102b, 102c may implement radio technologies such as IEEE 802.16 (i.e., Worldwide Interoperability for Microwave Access (WiMAX)), CDMA2000, CDMA2000 1X, CDMA2000 EV-DO, Interim Standard 2000 (IS-2000), Interim Standard 95 (IS-95), Interim Standard 856 (IS-856), Global System for Mobile

communications (GSM), Enhanced Data rates for GSM Evolution (EDGE), GSM EDGE (GERAN), and the like.

[0039] The base station 114b in FIG. 1A may be a wireless router, Home Node B, Home eNode B, or access point, for example, and may utilize any suitable RAT for facilitating wireless connectivity in a localized area, such as a place of business, a home, a vehicle, a campus, and the like. In one embodiment, the base station 114b and the WTRUs 102c, 102d may implement a radio technology such as IEEE 802.11 to establish a wireless local area network (WLAN). In another embodiment, the base station 114b and the WTRUs 102c, 102d may implement a radio technology such as IEEE 802.15 to establish a wireless personal area network (WPAN). In yet another embodiment, the base station 114b and the WTRUs 102c, 102d may utilize a cellular-based RAT (e.g., WCDMA, CDMA2000, GSM, LTE, LTE-A, etc.) to establish a picocell or femtocell. As shown in FIG. 1A, the base station 114b may have a direct connection to the Internet 110. Thus, the base station 114b may not be required to access the Internet 110 via the core network 106.

[0040] The RAN 104 may be in communication with the core network 106, which may be any type of network configured to provide voice, data, applications, and/or voice over internet protocol (VoIP) services to one or more of the WTRUs 102a, 102b, 102c, 102d. For example, the core network 106 may provide call control, billing services, mobile location-based services, pre-paid calling, Internet connectivity, video distribution, etc., and/or perform high-level security functions, such as user authentication. Although not shown in FIG. 1A, it will be appreciated that the RAN 104 and/or the core network 106 may be in direct or indirect communication with other RANs that employ the same RAT as the RAN 104 or a different RAT. For example, in addition to being connected to the RAN 104, which may be utilizing an E-UTRA radio technology, the core network 106 may also be in communication with another RAN (not shown) employing a GSM radio technology.

[0041] The core network 106 may also serve as a gateway for the WTRUs 102a, 102b, 102c, 102d to access the PSTN 108, the Internet 110,

and/or other networks 112. The PSTN 108 may include circuit-switched telephone networks that provide plain old telephone service (POTS). The Internet 110 may include a global system of interconnected computer networks and devices that use common communication protocols, such as the transmission control protocol (TCP), user datagram protocol (UDP) and the internet protocol (IP) in the TCP/IP internet protocol suite. The networks 112 may include wired or wireless communications networks owned and/or operated by other service providers. For example, the networks 112 may include another core network connected to one or more RANs, which may employ the same RAT as the RAN 104 or a different RAT.

[0042] Some or all of the WTRUs 102a, 102b, 102c, 102d in the communications system 100 may include multi-mode capabilities, i.e., the WTRUs 102a, 102b, 102c, 102d may include multiple transceivers for communicating with different wireless networks over different wireless links. For example, the WTRU 102c shown in FIG. 1A may be configured to communicate with the base station 114a, which may employ a cellular-based radio technology, and with the base station 114b, which may employ an IEEE 802 radio technology.

[0043] Referring now to FIG. 1B, a system diagram of an example WTRU 102 is shown. The WTRU 102 may include a processor 118, a transceiver 120, a transmit/receive element 122, a speaker/microphone 124, a keypad 126, a display/touchpad 128, non-removable memory 130, removable memory 132, a power source 134, a global positioning system (GPS) chipset 136, and other peripherals 138. It will be appreciated that the WTRU 102 may include any sub-combination of the foregoing elements while remaining consistent with an embodiment.

[0044] The processor 118 may be a general purpose processor, a special purpose processor, a conventional processor, a digital signal processor (DSP), a plurality of microprocessors, one or more microprocessors in association with a DSP core, a controller, a microcontroller, Application Specific Integrated Circuits (ASICs), Field Programmable Gate Array (FPGAs) circuits, any other

type of integrated circuit (IC), a state machine, and the like. The processor 118 may perform signal coding, data processing, power control, input/output processing, and/or any other functionality that enables the WTRU 102 to operate in a wireless environment. The processor 118 may be coupled to the transceiver 120, which may be coupled to the transmit/receive element 122. While FIG. 1B depicts the processor 118 and the transceiver 120 as separate components, it will be appreciated that the processor 118 and the transceiver 120 may be integrated together in an electronic package or chip.

[0045] The transmit/receive element 122 may be configured to transmit signals to, or receive signals from, a base station (e.g., the base station 114a) over the air interface 116. For example, in one embodiment, the transmit/receive element 122 may be an antenna configured to transmit and/or receive RF signals. In another embodiment, the transmit/receive element 122 may be an emitter/detector configured to transmit and/or receive IR, UV, or visible light signals, for example. In yet another embodiment, the transmit/receive element 122 may be configured to transmit and receive both RF and light signals. It will be appreciated that the transmit/receive element 122 may be configured to transmit and/or receive any combination of wireless signals.

[0046] In addition, although the transmit/receive element 122 is depicted in FIG. 1B as a single element, the WTRU 102 may include any number of transmit/receive elements 122. More specifically, the WTRU 102 may employ MIMO technology. Thus, in one embodiment, the WTRU 102 may include two or more transmit/receive elements 122 (e.g., multiple antennas) for transmitting and receiving wireless signals over the air interface 116.

[0047] The transceiver 120 may be configured to modulate the signals that are to be transmitted by the transmit/receive element 122 and to demodulate the signals that are received by the transmit/receive element 122. As noted above, the WTRU 102 may have multi-mode capabilities. Thus, the transceiver 120 may include multiple transceivers for enabling the WTRU 102

to communicate via multiple RATs, such as UTRA and IEEE 802.11, for example.

[0048] The processor 118 of the WTRU 102 may be coupled to, and may receive user input data from, the speaker/microphone 124, the keypad 126, and/or the display/touchpad 128 (e.g., a liquid crystal display (LCD) display unit or organic light-emitting diode (OLED) display unit). The processor 118 may also output user data to the speaker/microphone 124, the keypad 126, and/or the display/touchpad 128. In addition, the processor 118 may access information from, and store data in, any type of suitable memory, such as the non-removable memory 130 and/or the removable memory 132. The non-removable memory 130 may include random-access memory (RAM), read-only memory (ROM), a hard disk, or any other type of memory storage device. The removable memory 132 may include a subscriber identity module (SIM) card, a memory stick, a secure digital (SD) memory card, and the like. In other embodiments, the processor 118 may access information from, and store data in, memory that is not physically located on the WTRU 102, such as on a server or a home computer (not shown).

[0049] The processor 118 may receive power from the power source 134, and may be configured to distribute and/or control the power to the other components in the WTRU 102. The power source 134 may be any suitable device for powering the WTRU 102. For example, the power source 134 may include one or more dry cell batteries (e.g., nickel-cadmium (NiCd), nickel-zinc (NiZn), nickel metal hydride (NiMH), lithium-ion (Li-ion), etc.), solar cells, fuel cells, and the like.

[0050] The processor 118 may also be coupled to the GPS chipset 136, which may be configured to provide location information (e.g., longitude and latitude) regarding the current location of the WTRU 102. In addition to, or in lieu of, the information from the GPS chipset 136, the WTRU 102 may receive location information over the air interface 116 from a base station (e.g., base stations 114a, 114b) and/or determine its location based on the timing of the signals being received from two or more nearby base stations. It will be

appreciated that the WTRU 102 may acquire location information by way of any suitable location-determination method while remaining consistent with an embodiment.

[0051] The processor 118 may further be coupled to other peripherals 138, which may include one or more software and/or hardware modules that provide additional features, functionality and/or wired or wireless connectivity. For example, the peripherals 138 may include an accelerometer, an e-compass, a satellite transceiver, a digital camera (for photographs or video), a universal serial bus (USB) port, a vibration device, a television transceiver, a hands free headset, a Bluetooth® module, a frequency modulated (FM) radio unit, a digital music player, a media player, a video game player module, an Internet browser, and the like.

[0052] Referring now to FIG. 1C, a system diagram of the RAN 104 and the core network 106 according to an embodiment is shown. As noted above, the RAN 104 may employ a UTRA radio technology to communicate with the WTRUs 102a, 102b, 102c over the air interface 116. The RAN 104 may also be in communication with the core network 106. As shown in FIG. 1C, the RAN 104 may include Node-Bs 140a, 140b, 140c, which may each include one or more transceivers for communicating with the WTRUs 102a, 102b, 102c over the air interface 116. The Node-Bs 140a, 140b, 140c may each be associated with a particular cell (not shown) within the RAN 104. The RAN 104 may also include RNCs 142a, 142b. It will be appreciated that the RAN 104 may include any number of Node-Bs and RNCs while remaining consistent with an embodiment.

[0053] As shown in FIG. 1C, the Node-Bs 140a, 140b may be in communication with the RNC 142a. Additionally, the Node-B 140c may be in communication with the RNC 142b. The Node-Bs 140a, 140b, 140c may communicate with the respective RNCs 142a, 142b via an Iub interface. The RNCs 142a, 142b may be in communication with one another via an Iur interface. Each of the RNCs 142a, 142b may be configured to control the respective Node-Bs 140a, 140b, 140c to which it is connected. In addition,

each of the RNCs 142a, 142b may be configured to carry out or support other functionality, such as outer loop power control, load control, admission control, packet scheduling, handover control, macrodiversity, security functions, data encryption, and the like.

[0054] The core network 106 shown in FIG. 1C may include a media gateway (MGW) 144, a mobile switching center (MSC) 146, a serving GPRS support node (SGSN) 148, and/or a gateway GPRS support node (GGSN) 150. While each of the foregoing elements are depicted as part of the core network 106, it will be appreciated that any one of these elements may be owned and/or operated by an entity other than the core network operator.

[0055] The RNC 142a in the RAN 104 may be connected to the MSC 146 in the core network 106 via an IuCS interface. The MSC 146 may be connected to the MGW 144. The MSC 146 and the MGW 144 may provide the WTRUs 102a, 102b, 102c with access to circuit-switched networks, such as the PSTN 108, to facilitate communications between the WTRUs 102a, 102b, 102c and traditional land-line communications devices.

[0056] The RNC 142a in the RAN 104 may also be connected to the SGSN 148 in the core network 106 via an IuPS interface. The SGSN 148 may be connected to the GGSN 150. The SGSN 148 and the GGSN 150 may provide the WTRUs 102a, 102b, 102c with access to packet-switched networks, such as the Internet 110, to facilitate communications between and the WTRUs 102a, 102b, 102c and IP-enabled devices.

[0057] As noted above, the core network 106 may also be connected to the networks 112, which may include other wired or wireless networks that are owned and/or operated by other service providers.

[0058] Referring now to FIG. 1D, a system diagram of the RAN 104 and the core network 106 according to another embodiment is shown. As noted above, the RAN 104 may employ an E-UTRA radio technology to communicate with the WTRUs 102a, 102b, 102c over the air interface 116. The RAN 104 may also be in communication with the core network 106.

[0059] The RAN 104 may include eNode-Bs 150a, 150b, 150c, though it will be appreciated that the RAN 104 may include any number of eNode-Bs while remaining consistent with an embodiment. The eNode-Bs 150a, 150b, 150c may each include one or more transceivers for communicating with the WTRUs 102a, 102b, 102c over the air interface 116. In one embodiment, the eNode-Bs 150a, 150b, 150c may implement MIMO technology. Thus, the eNode-B 150a, for example, may use multiple antennas to transmit wireless signals to, and receive wireless signals from, the WTRU 102a.

[0060] Each of the eNode-Bs 150a, 150b, 150c may be associated with a particular cell (not shown) and may be configured to handle radio resource management decisions, handover decisions, scheduling of users in the uplink and/or downlink, and the like. As shown in FIG. 1D, the eNode-Bs 150a, 150b, 150c may communicate with one another over an X2 interface.

[0061] The core network 106 shown in FIG. 1D may include a mobility management entity gateway (MME) 152, a serving gateway 154, and a packet data network (PDN) gateway 156. While each of the foregoing elements are depicted as part of the core network 106, it will be appreciated that any one of these elements may be owned and/or operated by an entity other than the core network operator.

[0062] The MME 152 may be connected to each of the eNode-Bs 150a, 150b, 150c in the RAN 104 via an S1 interface and may serve as a control node. For example, the MME 152 may be responsible for authenticating users of the WTRUs 102a, 102b, 102c, bearer activation/deactivation, selecting a particular serving gateway during an initial attach of the WTRUs 102a, 102b, 102c, and the like. The MME 152 may also provide a control plane function for switching between the RAN 104 and other RANs (not shown) that employ other radio technologies, such as GSM or WCDMA.

[0063] The serving gateway 154 may be connected to each of the eNode Bs 150a, 150b, 150c in the RAN 104 via the S1 interface. The serving gateway 154 may generally route and forward user data packets to/from the WTRUs 102a, 102b, 102c. The serving gateway 154 may also perform other functions,

such as anchoring user planes during inter-eNode B handovers, triggering paging when downlink data is available for the WTRUs 102a, 102b, 102c, managing and storing contexts of the WTRUs 102a, 102b, 102c, and the like.

[0064] The serving gateway 154 may also be connected to the PDN gateway 156, which may provide the WTRUs 102a, 102b, 102c with access to packet-switched networks, such as the Internet 110, to facilitate communications between the WTRUs 102a, 102b, 102c and IP-enabled devices.

[0065] The core network 106 may facilitate communications with other networks. For example, the core network 106 may provide the WTRUs 102a, 102b, 102c with access to circuit-switched networks, such as the PSTN 108, to facilitate communications between the WTRUs 102a, 102b, 102c and traditional land-line communications devices. For example, the core network 106 may include, or may communicate with, an IP gateway (e.g., an IP multimedia subsystem (IMS) server) that serves as an interface between the core network 106 and the PSTN 108. In addition, the core network 106 may provide the WTRUs 102a, 102b, 102c with access to the networks 112, which may include other wired or wireless networks that are owned and/or operated by other service providers.

[0066] FIG. 1E is a system diagram of an example communications system 175 in which one or more disclosed embodiments may be implemented. In some embodiments, communications system 175 may be implemented using all or a portion of system 100 as shown and described with respect to FIG. 1A.

[0067] User device 180a, server 185, and/or service server 190 may communicate over communications network 195. These communications may be wireless, wired, or any combination of wireless and wired. Communications network 195 may include the internet 110, core network 106, other networks 112, or any other suitable communications network or combination of communications networks.

[0068] User device 180a may include a WTRU (such as WTRU 102a), or any suitable user computing and/or communications device such as a desktop

computer, web appliance, interactive television (ITV) device, gaming console (such as Microsoft XBOX™ or Sony Playstation™) or the like. User device 180a and/or applications executing on user device 180a may generate events such as mouse clicks, keyboard strokes, and the like. These events may be processed by user device 180a and/or may be transmitted to another device such as server 185 or service server 190.

[0069] Server 185 may include a web server, application server, data server, or any combination of these or other types of servers. Server 185 may include any suitable server device such as a server computer, personal computer, or the like. Server 185 may host applications accessible to user device 185a. For example, server 185 may include a gaming server hosting a massively multiplayer online game (MMOG), an email server, a web server hosting a website such as a social media website or blog, or other types of servers typically accessible by a user device over a computer communications network.

[0070] User device 180a may access server 185 over computer communications network 175 to interact with services that it provides. For example, user device 180a may access a game server hosted on server 185 to participate in a multiplayer online game. Access of server 185 by user device 180a may be via a client application executing on user device 180a or any other suitable mechanism. In some cases, the server 185 may receive events from user device 180a, or may send events to user device 180a. For example, the server 185 may send an event to user device 180a indicating that additional in-game resources are required for continued play.

[0071] Service server 190 may include a web server, application server, data server, or any combination of these or other types of servers hosted on a server device. Service server 190 may include any suitable server device such as a server computer, personal computer, or the like. Service server 190 may be configured to communicate with server 185, for example, over network 195 or any other suitable communications medium. Service server may be co-located with, combined with, or in direct communication with server 185.

[0072] Service server 190 may communicate with server 185 to provide services, such as third party services, to users of server 185. For example, a subscriber to a game hosted on server 185 may access server 185 from user device 180A and may subscribe to third party services for the game which are hosted on service server 190.

[0073] Service server 190 may be configured to receive and/or intercept events transmitted between user device 180a and server 185. For example, in some embodiments server 185 and service server 190 may be configured such that server 185 may send an event destined for user device 180a instead or additionally to service server 190, and service server 190 may send the event or another event, signal, or message to device 180a. For instance, in a case where server 185 includes a game server, server 185 may send an event to service server 190 indicating a requirement of a user of user device 180a, and server 190 may send the event or another signal or message to device 180a indicating that a resource is available to satisfy the requirement. In some embodiments, service server 190 may only forward the event to device 180a under certain conditions, such as based on a user preference and/or context information relating to the user of device 180a.

[0074] In some embodiments, the functions of service server 190 and server 185 may be implemented using the same device, or across a number of additional devices. In some embodiments, user devices 180b and 180c may communicate with server 185 and/or service server 190 via user device 180a. For example, user device 180a may forward a notification message from service server 190 to user device 180b via a peer to peer connection and may forward a notification message from service server 190 to user device 180c via network 195. In some embodiments, user devices 180a, 180b, and 180c may form a network, such as a peer-to-peer network, and such network may have a mesh topology, a star topology using user device 180a as a coordinating node, or any other suitable topology. In such embodiments, the peer-to-peer network may operate independently of server 185 and/or service server 190,

and may incorporate functionality that otherwise would be hosted by server 185 and/or service server 190, such as functionality described herein.

[0075] Everything that follows may, but is not required to be, employed and/or implemented using one or more, or part of one or more of the example systems discussed above. For example, in the context of a gaming session described herein, a user may play a game served by server 185 (a game server in this context) using user device 180a, and service server 190 may capture events from user device 180a and/or server 185 for purposes of profile stylization. For example, server 185 or service server 190 may implement a profile management system and/or data analytics system as further discussed herein. Not all components may be used; for example, service server 190 may be omitted and such functionality may be incorporated into server 185.

[0076] FIG. 2 is a schematic diagram illustrating an example service action control system 200. Service action control system 200 includes an active CoI object model 210 and service architecture 220.

[0077] Service architecture 220 may include a service server 230. In one example, service server 230 is implemented as a service server such as service server 190 as shown and described with respect to FIG. 1E. Service server 230 includes a service action controller 240, an operation dashboard 250, and a shared active object repository 260. In some implementations, functions associated with particular components of service architecture 200, such as service server 230 and/or its constituent parts, may be separated and implemented using different or additional components. In some implementations, functions of separate components of service architecture 200 may be combined into a single component, or fewer components. In some implementations, various components and functions may be omitted, or duplicated.

[0078] The service server 230 receives events from users and devices 270, and transmits service actions to users and devices 270. Service server 230 also receives commands from administrative programs 280, which may include an administration GUI of service server 230, or commands received

from other devices over a communications network via a suitable API. The service server 230 also interacts with application service functions 290.

[0079] Service server 230 includes suitable memory and processing circuitry configured to implement CoI object model 210. In this example, the CoI object model 210 is implemented on service server 230 as software stored in memory and executing on a processor of service server 230. Service server 230 interacts with users and devices 270, administrative programs 280, and service functions 290 based on the CoI object model. For example, various examples of users and devices 270 are modeled as User₁, User₂, User₃, User_n, and Device₁, Device₂, Device₃, Device_x in CoI object model 210. Various administrative programs 280 are modeled as App₁, App₂, App₃, App_N.

[0080] The CoI object model 210 and service server 230 may be implemented, for example, using service server 190 as shown and described with respect to FIG. 1E. User₁, User₂, User₃, User_x may include any suitable user devices, such as User device 180a as shown and described with respect to FIG. 1E. Device₁, Device₂, Device₃, Device_x may be Internet of Things (IoT)-type sensors, or any other type of sensors, which may supply event data directly or indirectly to the service action controller 240 (e.g., via communications network 195 as shown and described with respect to FIG. 1E). App₁, App₂, App₃, App_N may be administration applications associated with CoI object model 210, such as a command program, possibly including a GUI or other interface, executing on the service action controller 240, or executing on other devices and communicated to service server 230 using a suitable API (e.g., via communications network 195 as shown and described with respect to FIG. 1E).

[0081] The CoI object model 210 tracks various contexts of interest. For example, CoI C₁₁ may reflect a particular state of one or more variables, such as a progression of a user activity, environmental conditions, or any other suitable context information. CoI object model 210 also illustrates various events Events₁, Events₂, and Events₃. These events may reflect any suitable events, such as those described herein. For example, Events₁ may include a

user interaction with a program executing on user device User₃, such as a mouse click. This interaction may generate Events₁, which may be communicated to or otherwise captured or intercepted by the service server 230, and processed using CoI model 210.

[0082] In this example, CoI model 210 may detect a current context of interest C₁₁ for User₃. CoI model 210 may detect a change in context from context of interest C₁₁ to context of interest C₁₂ based on having received Events₁. For example, Events₁ may include a change in a context variable which causes the context to change from C₁₁ to C₁₂.

[0083] Following the change in context of interest from C₁₁ to C₁₂, service server 230 may receive Events₂ from Device₂. Events₂ may reflect a change in an output variable from an environmental sensor, such as temperature, for example. Events₂ may be communicated to or otherwise captured or intercepted by service server 230, and processed using CoI model 210. CoI model 210 may detect the change in context of interest from C₁₂ to C₁₃ based on having received Events₂. For example, Events₂ may include a change in a context variable which causes the change from context of interest C₁₂ to context of interest C₁₃.

[0084] CoI object model 210 also includes various composite contexts of interest CoI₁, CoI₂, CoI_i. Each of CoI₁, CoI₂, CoI_i corresponds to a defined sequence of contexts, and may be associated with a corresponding AEP (in this case, AEP₁, AEP₂, AEP_i) as further discussed herein. It is noted that in some implementations, non-compound contexts of interest may be associated with a corresponding AEP. CoI₁ in this example corresponds to the sequence of context of interest C₁₁, followed by contexts of interest C₁₂ and C₁₃. Likewise, CoI₂ corresponds to the sequence of a context of interest C₂₁ followed by a context of interest C₂₂. Similarly, CoI_i corresponds to a sequence of a context of interest C_{i1}, C_{i2}, C_{i3}, and C_{i4}. In the example above, starting from context of interest C₁₁, the example sequence of Event₁ and Event₂ causes the context of interest tracked by CoI object model 210 to change from C₁₁ to C₁₂, to C₁₃.

This series of changes corresponds to compound context of interest CoI_1 , as shown in FIG. 2, and the CoI_1 is identified by CoI object model 210.

[0085] CoI object model 210 also includes service actions $Actions_1$, $Actions_2$, $Actions_3$. These service actions are defined in CoI object model 210 to correspond to various contexts of interest. For example, $Actions_1$ are defined to correspond to CoI_1 , $Actions_2$ are defined to correspond to CoI_2 , and $Actions_3$ are defined to correspond to CoI_3 . The CoI object model 210 may trigger the various service actions, directly or indirectly, in response to identifying a corresponding context of interest or compound context of interest. For example, $Actions_1$ may be transmitted to $Device_1$ by service server 230 in response to CoI_1 having been identified by CoI object model 210 (e.g., as included in an appropriate AEP).

[0086] Similarly, $Actions_2$ may be transmitted to $User_2$ by service server 230 in response to CoI_2 having been identified by CoI object model 210 (i.e., by having detected a change in context of interest from C_{21} to C_{22}) and $Actions_3$ may be transmitted to $Device_N$ in response to CoI_i having been identified by CoI object model 210 (i.e., by having detected a change in context of interest from C_{i1} to C_{i2} , to C_{i3} , to C_{i4} in response to $Events_3$ having been received from $Device_3$).

[0087] The various contexts of interest and/or compound contexts of interest are defined for CoI object model 210 in a memory of service server 230 by an insert action from one or more of the administrative programs 280. This is illustrated with respect to CoI object model 210 where compound context of interest CoI_i is defined using an insert action from App_3 . Here, an app App_3 defines compound context of interest CoI_i by insert action 230 via any suitable mechanism, such as those discussed herein. For example, an administrator may subscribe App_3 to the service action control system 200, which may be hosted on a third party server for example, and may perform insert action 230 by transmitting a message from a client device to the third party server via a suitable API or other interface.

[0088] Service action controller 240 includes one or more controllers for receiving and comparing input events against the CoI objects, and coordinating service control actions as further discussed herein.

[0089] Operation dashboard 250 is a display for displaying various information, such as summaries of service actions, CoIs, and/or activities as further discussed herein. Operation dashboard 250 may be local to service server 230 or accessible remotely via a suitable computer communications network.

[0090] Shared object repository 260 is a memory or combination of memories which may be local to service server 230 or remote and in communication with service server 230 over a computer communications network. The shared object repository stores objects such as CoIs, AEPs, and event and state variables involved in detection of CoIs and execution of service functions 290 for example, as further discussed herein.

[0091] Various embodiments described herein, such as service action control system 200, may improve the efficiency of context aware event processing and the effectiveness of service actions offered by multiple service providers to the same set of users and devices in a multi-service provider pervasive service environment. In an embodiment, Context of Interest (CoI) patterns may be used to represent the occurrences of behavior patterns of activities in IoT environment (e.g., shopping, working, exercising, playing) that are of interest for one or more service providers to offer one or more context aware service actions.

[0092] Embodiments described herein may support one or more cooperative service entities. Each cooperative service entity may provide a cooperative action control service to one or more service providers from one or more companies. Each cooperative service entity may use cooperative service action descriptors (provided by the proposed system to one or more service providers) to enter cooperative objectives such as cost and effectiveness objectives for a set of cooperative service actions. Examples of cost and effectiveness objectives may be operation expenses and revenue generated.

[0093] In an embodiment in which the cooperative service entity is a single company with multiple participating organizations that provide the cooperative services, the action descriptor may be managed by an administrator of the cooperative service entity of the company. For service providers from independent companies, the action descriptor may be managed by an administrative entity that is independent from the participating companies. Depending upon the types of services, the cost and effectiveness definitions may be proprietary or shared between independent service providers.

[0094] In an embodiment, a cooperative service administrator may manually inspect the property of an IoT device resource, and may target a user with respect to each action types. If a device or a target user only supports one specific type of action at a time, embodiments may sequentially schedule the action type based on priority. If the actions are conflicting at a semantic level that needs to be reordered, or are mutually exclusive such that one of the actions need to be deleted, the cooperative service administrator may notify the owner of the actions to mediate resolution external to the system.

[0095] When there are conflicts that cannot be easily detected by the administrator and/or the service providers in the one or more companies, embodiments may support automated adjustment of priority, which defines the scheduling order, based on cost and effectiveness measurements. The cost and effectiveness measurements may be designed by administrators of the cooperative service entities, possibly with input from participating service providers, and may be enforced as cooperative policy among the participating service providers. The policy definition may be CoI specific and may be implemented as a cooperative action control policy service entry point that takes the action descriptor as input to generate priority for each action accordingly.

[0096] The cost and effectiveness measurements may be based on business performance indicators defined by one or more companies. In the

case that all service actions are provided by one single company, the cooperative service entity of the company may define and manage the cost and effective measurements policy. In the case of multiple independent companies, the cost of action may be defined based on a “service fee” required by the service entity and agreed by participating service provider. The effectiveness may be measured by customer feedback or by other automated tracking methods. In an embodiment, the automated tracking may be based on a conversion rate that defines a percentage of customers that purchased a product after receiving a sale promotion.

[0097] In an embodiment, the CoI pattern matching may be implemented by a set of Actionable Event Patterns (AEPs) matching functions. Each AEP may include a unique identifier, a sequence of event patterns, and an action list. The cooperative service action control system 200 shown in FIG. 2 may manage the action list and schedule the execution of actions in the list. The actions in the AEP action list may contain service entry points to invoke external services (e.g., via web service application programming interfaces (APIs) or IoT service APIs). In an embodiment, the AEPs may match a sequence of event patterns against the pattern defined in CoI patterns. The AEPs may then trigger the execution of a list of actions when detecting the occurrences of input events matching the one or more CoI patterns.

[0098] The AEPs may also generate one or more intermediate events. The one or more intermediate events may change the states of the complex event pattern matching processes and/or data stored in a working memory or database. The AEPs may be organized hierarchically to process events of interests from multiple low level devices as well as high level complex context change events. The AEPs may also store and access user and device behavior information from a data repository.

[0099] In an embodiment, the CoI patterns and the action list defined in the AEPs may be shared with one or more service providers. After a service provider is authorized to join the cooperative service, the service provider may

select the CoIs and may insert new service actions to the service action list of the AEP associated with the CoIs. The service actions may invoke execution of new service functions implemented by the service providers in the same server or in remote servers.

[0100] Each CoI may be defined and implemented by a shared Active Object class. Applications may instantiate a CoI object class and define the CoI patterns. The Active Object class may also contain an AEP function to detect the event patterns and trigger actions in the action list. The following is an example of a high level data model for CoI and action.

[0101] The AEP may be defined as a method to detect event patterns and trigger actions. The AEP may include descriptors such as, but not limited to: Name, which may be is a descriptive name for the AEP; ID, which may be a unique identifier for the AEP; Effective time, which may be the time that defines when the AEP is or will be activated and when it will be deactivated; Effective spatial region; CoIList, which may be a list of CoI event objects which is used to detect CoI pattern; CoIPattern; ActionList, which may be a list of actions triggered in response to detection of the patterns in CoIPattern; Creation time, which may indicate the time when the AEP was created using, for example, an Epoch time stamp; and Creator, which may be the administrator of the service provider who created the AEP. It should be noted that the AEP may be created by the information technology (IT) group of a company or individual user, using the cooperative service administration GUI and APIs provided by the proposed system function as described in more detail below.

[0102] The CoIPattern may be a high level event detection pattern that defines the occurrences of a sequence of event patterns. Each of the event patterns may be a CoI event generated when the CoI pattern is detected. For example, a CoI event may be detecting a VIP customer at an entrance. Another CoI event may be detecting the VIP waiting for service for a long time (e.g., over 2 minute). Another CoI event may be detecting that the VIP is frustrated and leaving the shop without purchasing any items or services. The

CoI pattern in AEP can be combinations of the one or more occurrences of CoI events in time sequence with specific temporal relationship. It may also be a simple combination of occurrence of event without specific temporal order. An example of the CoIPattern is given in an example provided below. In the example, the administrator may define CoIPattern as simple conjunctive (e.g., ANDing) of multiple occurrences of CoIs. More complex composition based on location and time can also be supported.

[0103] The CoI may include attributes such as, but not limited to: ID; Name; User ID; Device ID; Application ID; Activity ID; Activity type; Environment condition ID; Environment variables; Location; Effective-time (i.e., a time window that the CoI may be activated, such as, for example store hours); and Service action list which may link to service actions depending on the CoI.

[0104] In an embodiment, CoI ID may be used for a CoI event that signifies the detection events from one or more devices. For example, the sales organization may be interested in CoI generated from an access control system or face recognition system if a VIP or a person in black list has entered the premises. Activities may define a customer entering a building, placing an item in a shopping cart, or engaging in intensive exercise. The Environment condition may be temperature and/or occupancy and/or a place or location.

[0105] The CoI may consist of events from multiple devices and/or application services in a hierarchy of CoI. For example, the CoIs for a shop entrance may be defined to detect lower level events such as customer entering the store, picking up a smart shopping cart, and providing a user ID and shopping list to the shop. In this case, the CoIs may consist of a list of events. The entrance CoI may generate a CoI event to signify that a customer arrived and is ready for the smart shopping services. CoIs may be grouped together to form a list of CoIs. The list of CoIs may be associated with a CoI pattern that detects the sequence of occurrences of the CoIs occurring in a temporal and/or spatial relationship.

[0106] Each action in the ActionList may include attributes such as, but not limited to: Action descriptors such as ID, Priority, Cost, EffectivenessMeasurement (e.g., return on investment (ROI) benefits or revenues), IoTResourcesList (GroupID), and TargetedUsersList (GroupID); ServiceEntryPoints for invoking the service functions; ServiceProviderID; ServiceDescriptor; and SecurityTokens, which provide options to support secure and anonymous actions.

[0107] In an embodiment, the cost may be defined by the administrator of one or more cooperative service providers and verified by the administrator of cooperative service entity. The cost may be adjusted by administrators manually or automatically by an external cost between the cooperative service participants and the administrative service entity.

[0108] EffectivenessMeasurement may contain a function that can be used to compute the effectiveness of the particular action after it is taken, e.g., compute the cost/benefit of the action after it was taken, or compute its return on investment. The function is computed after the action is completed by the service participants.

[0109] The IoTResourcesList may contain multiple IoT devices that may be involved by each action in each entry of the action descriptor. The IoT resource may further contain information about the IoT devices such as if the device may only handle one action at a time or if it can support multiple concurrent actions.

[0110] The TargetedUserList may contain information about the user or user group that is expected to receive the action in the form of messages or through other IoT devices (e.g., temperature sensors and control devices, motion sensors, inventory sensors, automated dispensers, automatic locks controllers, automatic lights controller, automatic sprinkler system controllers, or speakers and displays) near the user's location. It may also provide information about the user preference (e.g., do not disturb or privacy protection indicators).

[0111] In an embodiment, the action in the action list may be executed based on priorities defined in the action descriptor. The following example describes an example of execution policies that may be used. Order the action in the action list based on priority. Schedule the action execution based on the action list order. If the IoTResourceList or the TargetedUserList are blocking (e.g., do not support concurrency), then the execution is blocking. Actions may be executed in sequence. If the IoTResourceList or the TargetedUserList are non-blocking, then, the actions may be executed non-blocking (i.e., executed asynchronously without waiting). If the actions to an IoT resource are mutually exclusive (e.g., set the same thermostat temperature to two different settings by two different service actions), the low priority action may be deleted and the target users and service providers may be notified.

[0112] Embodiments may control the execution of a set of actions based on the priority attributed to them in the action descriptor. The priority of the action may be adjusted based on a cooperative objective and the effectiveness measurement of each action (e.g. effectiveness to a user or a group of users) over a pre-defined time window. Using this mechanism, the priority of the action may adapt with the relative effectiveness measurement. The priority of actions with high effectiveness measurements may increase incrementally and they may eventually be executed first. Ineffective actions may eventually be phased out or inhibited. As an option, the service action activation may be protected by secure access tokens.

[0113] The system shown in FIG. 2 may include function modules and interfaces to support the system administration, data repository access, operation dashboard, and the context aware service action controllers.

[0114] The system administration may include graphical user interfaces (GUIs) and service application programming interfaces (APIs) for users, devices, and applications to participate in selected cooperative services. The GUIs and APIs may be provided to support mechanized administration steps for the administrator of service entity and service providers from one or more organizations. The administration steps may include: defining CoIs (e.g.,

based on the example CoI class object presented above), observing existing CoIs, defining service actions (e.g., based on the Action descriptor as defined above, defining AEPs (e.g., based on the example AEP class object presented above) capable of detecting sequences of CoIs, and inserting service actions developed independently into one or more AEP action lists.

[0115] The administration functions may be categorized into a number of function groups, described in detail below. One example of a function group may be CoI and AEP GUIs and APIs. The CoI pattern definition and detection may be based on event pattern detection logics implemented using event processing system. In an embodiment, the system may provide GUI templates for operators of service organizations to define contexts of interests. A typical CoI definition GUI may be a template containing the event patterns that match a sequence of input events. When a CoI pattern is detected, a CoI event may be generated and used by an AEP and action list, which may be provided to operators via a GUI template. When an AEP matches a sequence of CoI event patterns, a list of actions may be invoked in parallel. It should be noted that the detection logic may simply detect occurrences of all the CoIs or occurrences of CoIs that match temporal sequences or both. When there are conflicts in resources (e.g., actions on a device that can only handle one action at a time), the action may be scheduled based on priority. Actions with higher priority may be processed first.

[0116] Another function group may be service action management GUIs and APIs. In an embodiment, service providers may define the CoIs and may share the CoIs with other service providers through a data repository. In an embodiment, the system may provide templates to allow operators from service provider organizations to view and provision a set of CoI definitions and a set of independently developed and deployed service actions. GUIs or APIs may be provided for the operator to insert service action entry points to the action list for the set of CoIs. The service providers may also observe the CoI patterns in existing AEPs defined by other service providers and may insert the service actions to the action list of the AEP.

[0117] Another function group may be certification process GUIs and APIs. The cooperative service entity of the proposed system may inspect the service actions inserted (or requested to be inserted) by service providers or operators and may authorize the insertion after the service action is certified and tested by a certification process. The certification process may include methods to decide the cooperative objective and methods to calculate priority based on the objectives. The administrator of the cooperative service entity may check if there are high level business process conflicts (e.g., offering clearance items after an up-sale item was sold). Additional functions may be provided to service provider or operator or multiple companies to perform independent processes that may optimize the cost and effectiveness parameters defined in the action descriptors. For example, a company may opt to pay more for the action than others when the company observes the benefits of the actions through effectiveness measurements.

[0118] Operation dashboard 250 may track the participating application and service identities and may report effectiveness measurements defined in the cooperative objective with respect to the actions launched by each application.

[0119] A shared active object repository may store the CoI objects, input events, behavior metrics, output actions, activities, environment conditions, cooperative objectives, effectiveness measurements, and state variables which support the pattern detection methods. Administration data may also be stored in the repository. The active object repository may support fast searching of CoI patterns based on location, time, and activities. The CoIs for each user or user group may also be activated when detecting the presence of the user. History information of user activities, actions, and effectiveness measurements may also be stored in the repository. The active object repository may contain the description of CoIs and AEPs that can be searched or referenced by service providers based on descriptors defined for each of the CoIs. For example, a CoI may have multiple key words and properties specifying the location, activities, effective time window, effective date, and

context specific attributes (e.g., entering the perimeter, exiting the perimeter, passing an aisle, picking up an item from a shelf, placing an item in shopping cart, and being identified as a VIP). Each AEP may also have properties about the CoIs and patterns of CoIs that may be searched and displayed by service providers.

[0120] A set of service action controllers may process input events against the patterns defined in CoI objects and may coordinate service control actions from multiple cooperative services. The events may contain input and state information of users and devices collected from IoT sensors. When patterns are matched, a list of service actions may be executed based on priority of the action. When the CoIPatterns are defined and used by many service providers, it may remain unchanged unless all service providers are no longer interested in the CoI pattern and there are no actions in the action list. The AEP may only need to link to the CoIPatterns or embedded in the specific CoI active objects that consist of all the CoIs and associated CoI Pattern.

[0121] Embodiments may support the dynamic creation of CoIs, Actions, and AEPs using an active object model. Objects may be created and linked together to allow service providers to insert CoIs, Action service entry points, and AEPs dynamically. Furthermore, embodiments may support AEP action list sharing, which may allow multiple actions to be added dynamically after a new AEP is created.

[0122] As shown in the conceptual object model in FIG. 2, more than one CoI may be grouped together to form a composite sequence of patterns. Each composite sequence may have one or more AEP detection methods. Each AEP may be associated with a CoI pattern that may include one or more CoI events that are generated when one or more device event patterns are detected. The CoIs, AEPs, and actions may run in servers, PCs, wearable devices, mobile phones, IoT devices with memory, CPUs, and communication resources. Service actions may be service calls and/or messages to alert a user, invoke service functions offered by multiple service providers, and perform actions on IoT devices. The service functions called by the service actions may be

modeled as service entry points. The service entry points and the service functions may be provided by one or more service providers. Service providers may be end users, companies, network service providers, cloud service providers, and/or other application service providers. One example of user that may be a service provider may be a “crowd sourcing service” among a community of users. Another example of a service provided may be a user who provides personal information from a mobile device or social network.

[0123] The service action function implementation may be decoupled from the CoI objects. Applications and service providers may dynamically create CoIs and may share the CoIs with other applications and services. Each application may inspect and search for CoIs among, for example, a set of provided and/or shared CoIs to optimize the effectiveness and timeliness of service functions. After identifying the CoIs, service providers may create service entry points and may insert the service actions to the action lists. The service actions in the action list may be triggered when input events match the patterns of selected CoIs.

[0124] The set of triggered actions may in turn trigger reactions from users or devices which may in turn result in new events to trigger more actions. The state variables defined in the events and cumulative behavior metric derived from the state variables may be recorded as part of the context variables of the active CoI objects. The context variables may be stored in CoI repository and may be dynamically manipulated using the methods defined in CoI object.

[0125] Embodiments of the system may include features such as efficient event pattern processing, effective service action coordination, and information sharing among streamlined business.

[0126] Embodiments may include a cooperative objective and service action administration feature. This may provide administration service for multiple service providers to participate in one or more cooperative service groups in one or more shared pervasive IoT environments. It may also provide templates and APIs for service providers to define service entry points for

independently defined value added service functions. It may also provide templates to define CoIs, and AEPs. It may also provide templates and APIs to support dynamic insertion and modification of service actions in action lists of selected CoIs and AEPs. It may define cooperative objects and may agree to the cooperative objective defined by the cooperative service entity of the shared IoT environment or service policy formulated by participants. An example of cost and effectiveness based cooperative objectives may be described in the cooperative action descriptor provided below.

[0127] Embodiments may include an integrated context aware event pattern processing feature. The system may reduce duplicated event processing steps by offering CoI detection services to participating cooperative service providers. The CoI patterns may be defined by one or more service providers and shared with other service providers who are also interested in providing services under the same CoI pattern dynamically. For example, when a shopper is looking around for items in a shop and the items have not been placed in a smart shopping cart, an item locator service may provide the locations of the items to the user. At the same time, the sale automation service may promote items of similar types for up-sale or inventory reduction.

[0128] Embodiments may enable dynamic insertions of service actions based on user behavior and context. Based on additional information about the user behavior profile (e.g., preference, interest, and routine tasks), each service function may dynamically insert or remove the actions from the action list of different CoI pattern detection AEPs. For example, shopping guide and special promotion service actions may be added to the ActionList of an AEP associated with new customers. Multiple waypoint detection points may be added for new visitors. Additional surveillance tracking and recording service action may be added on detecting a customer is wandering in the store.

[0129] Embodiments may coordinate and improve service action effectiveness. The system may analyze the priority of the service actions based on effectiveness measurement criteria defined in an action descriptor. Based on the effectiveness measurement, the system may prioritize the order

of service actions from one or more service providers. For service actions that result in low effectiveness measures, the service action may be assigned to a lower priority. For example, effective up-sale actions, which generate more revenue than ineffective clearance sale actions, may be assigned with higher priority and scheduled before clearance sale actions.

[0130] Embodiments may include a shared active object repository to streamline business processes. The system may support the creation, update, and execution of CoI patterns and service actions using an active shared object model. The event and state variables involved in CoI detection and action execution functions may be tracked in the active object repository. The objects stored in the repository may be accessible by all the participating services with proper access control and protection. Customer service employees or automated systems may use information on shopper purchase behavior patterns from a customer database, as well as the purchased item list from shopping carts, to promote new products among cooperative service providers or organizations. Entrance and exit records collected by security access control system may be used by employee attendance systems and occupancy detection services. Access control and surveillance camera services may also provide valuable customer behavior and environment conditions information for applications to decide when and where to engage the customers and provide personalized services.

[0131] Methods and systems for controlling cooperative retail shopper management services may be described in detail below. In a first example, service functions may be provided to users (e.g., shoppers) at different moments of engagements synchronized by the occurrences of CoI in IoT enabled stores. IoT sensor events may include variables such as shopper locations, items in a smart shopping cart, and shopping list and service functions offered by multiple retail business operation management systems.

[0132] Embodiments may include tracking measurable cooperative objectives that may be defined and agreed by participating service functions. A cooperative service entity who offers the cooperative service to multiple

service providers may define and develop cooperative objectives based on business objectives related to the types of IoT devices and the context of service. When designing the cooperative objectives, the service entity may consult or collect input from multiple service providers. A set of objects that may be in common to multiple services may include: revenue, including usage of coupons and/or additional items purchased; profit, including up-sale of high margin products; efficiency, including energy saved or incidence resolved; and cost, which may include the cost of each service action.

[0133] Embodiments may include defining and detecting evolving CoIs, labeled as C_i , during a shopping activity. Events from multiple IoT devices and service platforms may be collected. Each CoI may represent the state of progression of a user's activities detected by IoT sensors.

[0134] For example, CoI event C_1 may be when a shopper X, enters a store, allocates a shopping cart X, and submits a shopping list in a mobile app to a shopping cart and/or other customer support and sale automation applications of the store. It should be noted that this CoI may collect events from multiple devices or systems and generate a CoI event C_1 . CoI event C_2 may be when shopper X has a purchase-history {Count, Item, Time} (e.g., 2 dozen eggs per week) in a shopping application of the store. CoI event C_3 may be when the shopper X is passing the shelves for {Items} in the shopping list (e.g., eggs, others), or is asking for help for an item U that is not in the shopping list. CoI event C_4 may be when the shopping cart X has "not" detected {Items, count} in shopping list or purchase-history. CoI event C_5 may be when the shopping cart X prompts shopper X to subscribe and opt-in to a reminder service and shopper X interacts with shopping cart X to complete the subscription and opted in the service. CoI event C_6 may be when a sales person Z is notified to help shopper X. CoI event C_7 may be when the sales person Z receives shopper X profile and shopping list. CoI event C_8 may be when the sales person Z provides the shopper X with the item location. CoI event C_9 may be when sales person Z recommends shopper X one or more up-sale items. CoI event C_{10} may be when shopper X purchases the up-sale item.

CoI event C_{11} may be the identification of shopper X as, for example, a new user, a VIP customer, or a black list customer. The identification may use, for example, an access control system or a camera-based face recognition system.

[0135] Embodiments may collect the CoI events from multiple IoT devices, extract variables defined in the CoIs, and detect the occurrences of the event patterns that match each CoI. When the sequences of occurrences of one or more CoIs match the pattern defined in COIPatterns, the list of actions may be triggered. For example, the system may track and match events related to shopper activities modeled by multi-dimensional context parameters, thresholds, time, locations, and other conditions defined in the CoI events C_1 to C_{11} . It should be noted that the pattern matching may be implemented in the methods inside of a CoIPattern object or a separate AEP object that has access to the CoIs in the CoIList and can evaluate the CoI events using the CoIPatterns.

[0136] Service providers may develop service functions, deploy the service functions, and insert the service entry points into action lists as actions (denoted as A_i) of AEPs. When an AEP detects a CoI pattern, the list of service entry points in the action list may be triggered. The set of CoI events (denoted as $C_1, C_2, \dots$) in the AEP may be passed on as parameters to the service entry points that implements context aware service functions.

[0137] Action A_1 may be invoke (ShopperProfileSRV, C_{11}). This may get a profile of a shopper using identification of the shopper. For example, a ShopperProfileSRV service entry point may consist of service functions, such as access customer database, to obtain customer profile that contains purchasing behavior patterns. Action A_2 may be invoke (ReminderSRV, C_1, C_2, C_3, C_4). This may be a for missing items reminder. Action A_3 may be invoke (PromotionSR, C_3, C_4, C_5). This may be for a coupon on popular items. Action A_4 may be invoke (RecommendationSRV, C_9, C_{10}). This may be for cross/up-sale items. The recommendation service entry point may consist of service functions required to provide recommendations based on user current

shopping context (e.g., the item placed in shopping cart) and user purchasing history.

[0138] Action A_5 may be invoke (ReviewSRV, C_3). This may be for product rating. Action A_6 may be invoke (ConversionAlert, C_4). This may be for a sales conversion rate. The sales conversion rate service entry point may consist of functions to access real-time transaction records and promotion records to obtain the percentage of customers who actually purchased the product after a sale engagement (e.g., from a sale person or from a promotion coupon).

[0139] Action A_7 may be invoke (VIPService, C_{11}). This may insert multiple CoIs for a VIP shopper. Action A_8 may be invoke (NewCustomerService, C_{11}). This may provide an identity to a new customer service. Action A_9 may be invoke (face recognition feature, C_1). This may activate face recognition features from a surveillance camera to detect customer identity (e.g., VIP status, new customer, or black list). This action may be used to detect C_{11} . Action A_{10} may be invoke (customerID detection, C_1). This may use the access control system to detect employee and/or shopper identities such as sales, security guard, VIP shopper, new customer, or black list. This action may be used to detect C_{11} . Action A_{11} may be invoke (dispatch security guard, C_1). This may dispatch a security guard when a black listed customer or intruder is in the shop.

[0140] Service providers may create AEP templates (denoted AEP_i) having the format: “ AEP_{ID} : (CoI patterns), Then, Do {Action list}.” The AEP for a sequence of CoIs may be implemented as methods for a high level objects representing the sequence of CoI patterns. It should be noted that the pattern may contain additional predicates for evaluating the detailed contents of entities in the CoI patterns. For example, the following AEPs may detect the occurrences of different sets of CoI events in a simple conjunctive form (ANDing).

[0141] AEP template AEP_1 may be AEP_1 : If (C_1), Then, Do { A_9 , A_{10} }. This may trigger a shopper identification service using surveillance and/or

access control services. AEP template AEP₂ may be AEP₂: If (C₁, C₂, C₃, C₄), Then, Do {A₂}. This may be a missing items reminder. AEP template AEP₃ may be AEP₃: If (C₁, C₂, C₃, C₄, C₅), Then, Do {A₃, A₄, A₅}. This may be a promotion service. AEP template AEP₄ may be AEP₄: If (C₆, C₇, C₈, C₉, C₁₀), Then, Do {A₆}. This may be for sales force management. AEP template AEP₅ may be AEP₅: If (C₁ and C₁₁ (ShopperX = "VIP")), Then, Do {A₇}. This may be for VIP service. AEP template AEP₆ may be AEP₆: If (C₁ and C₁₁(ShopperX = "new customer")), Then, Do {A₈}. This may be for new customer support. AEP template AEP₇ may be AEP₇: If (C₁₁ ("ShopperX = BlackListedCustomer")), Then, Do {A₁₁}. This may be to dispatch a security guard.

[0142] Embodiments may execute actions in the action list of triggered AEPs. When there is a conflict between two or more actions on targeted users or devices, a conflict resolution policy may be applied based on priorities assigned to each action. The priority may be defined based on criteria with respect to the cooperative objective and cost functions associated with each action. The system may schedule the actions in such a way that low priority items may be executed within a maximal time delay to prevent starvation (e.g. to allow a low priority item to be executed often enough so that its effectiveness may continue to be measured).

[0143] The cooperative objectives may be defined and managed by a service administrator. Furthermore, the priority may be defined based on both the cost and the effectiveness. As illustrated in Table 1, the priority may be a function of both the cost that the service provider is willing to pay, and the effectiveness measured by user feedback and/or conversion rate (e.g., the percentage of customers who purchased the product after receiving the sale promotion). This may be monitored by the cooperative service entity or provided by service providers from different organizations. The cost attributes may be derived from both the actual cost of service and/or the fee that the service provider may have paid for each service action. If a service is not effective from user's point of view or from the ROI measured by the company or service provider, the effectiveness attribute may be decreased, which will

bring the priority down as well. Even if the service provider is willing to pay for the service action (e.g., promotion at high cost), the priority for the service action may decrease unless the effectiveness is maintain or improved.

Action ID	App ID	Service Provider ID	Priority	Cost per action	Effectiveness, ROI	IoT sensor list	Action Target: Users / devices
A ₁	App1	SP1	3	0.2	20%	IoT 1	U1, U2
A ₂	App4	SP1	5	0.5	10%	IoT 2, IoT3	device-Thermostat
A ₃	App3	SP2	5	1.0	5%	IoT 3, IoT5	Any User
A ₄	App7	SP3	7	10	3%	IoT 1, IoT 7	VIPs
A ₅	App8	SP3	9	5	20%	IoT 8	Users of one or more services

Table 1

[0144] Service providers may observe the priority level setting and find that the priority level oscillates. The oscillation may be caused by the changing cost or effectiveness. Oscillations that are caused by frequent cost adjustment may cause a warning signal to be generated to alert the administrator. More weight may be assigned to effectiveness to improve customer satisfaction and business efficiency.

[0145] Other service action scheduling methods may also be provided based on application specific contexts. For example, variable round robin or other probabilistic scheduling mechanisms can be adopted to ensure that all actions will be scheduled based on allowed time or other resource measurements.

[0146] Table 1 also shows an example of action descriptors related to cooperative objective definitions, such as ROI and cost of actions, associated with a list of IoT sensors. Priority level may be categorized and ranked by

simple multiplication of cost and effectiveness ROI. Additional criteria such as severity level of incident or security alert level may also be defined in the action descriptors for other types of activities or applications. The administrator may monitor the action descriptors to ensure that the actions are legitimate and can be verified with the corresponding participants providing the service actions. The actions may be applied to a subset of users defined in the Target Users column. The actions may be restricted to one or more users, user groups, and users of specific service specified in the Target Users column of the action descriptor.

[0147] FIG. 3 is a system diagram illustrating an example embodiment of a service control system 300. Service control system 300 may include IoT sensors such as location sensor 305, item radio frequency identification (RFID) reader 310, and shopper behavior sensor 315 for aisle, shelf and shopping card locations.

[0148] The IoT devices may be mapped to one or more virtual three-dimensional (3D) environments that support inventory and location tracking using virtual objects and one or more bots to model shelves, items, shopping carts, and shoppers. Such environments may be hosted on a service server 320, which may be implemented using service server 190 as shown and described with respect to FIG. 1E for example. In this context, a bot is a software application executing on a service server such as service server 320. Service server 320 is in communication with location sensor 305, item RFID reader 310, and shopper behavior sensor 315 via communications network 325. Communications network 325 may be any suitable computer communications network, such as communications network 195 as shown and described with respect to FIG. 1E for example.

[0149] A bot may be assigned to each shopper and may move in the virtual world locations synchronized with the physical location and orientation of the shopper in the store. In the example of FIG. 3, bots 330 and 335 are each assigned to an identified shopper. The bots may instantiate CoI objects for each user (shopper in this case) and may execute the AEPs associated with

the CoI objects. The CoI objects and AEPs may be stored in a shared contexts of interest repository 390, which may be hosted on service server 320 or remotely. Bots 330 and 335 may receive event data or other signaling transmitted from location sensor 305, item RFID reader 310, and/or shopper behavior sensor 315 (or any other available sensors) via communications network 325.

[0150] As shown in FIG. 3, bots 330 and 335 may interact with multiple types of virtual shopping service functions, such as functions for a customer profile 340, marketing and promotion 345, inventory loss and prevention 350, discount and price matching 355, calendar and reminder 360, product review and ranking 365, or any other suitable function. Such functions may be hosted on a second service server 370, which may communicate with service server 320 via communications network 325. Alternately, such functions may be hosted on the same service server 320, or using any other suitable topology.

[0151] Bots 330 and 335 may collect IoT shopper behavior event data including, for example, shopper ID, shopping list, and complex user behavior. Complex user behavior may include location, trajectory, gesture, and interactions with the items in the shelf or in shopping cart. Bots 330 and 335 may track the complex user behaviors and the environment conditions during shopping activities. Bots 330 and 335 may feed the collected event data to AEP methods, which may be processed in service server 320 to detect moments of opportunity to engage the users/shoppers for one or more application service functions.

[0152] The cooperative service control system 300 may support IoT service enablement and may streamline business process integration among multiple existing business and facility management systems.

[0153] In another embodiment, methods and systems for controlling service actions from multiple IoT enabled value-added service extensions of multiple business and facility management systems are disclosed. The management systems may include lighting control, energy management, camera surveillance, access control, and employee management services.

[0154] In an embodiment, events may be gathered from IoT, existing systems, and users. The events may include: geo-location events from geo-location services; access control events and security guard check point events from building security access control systems; waypoint object detection events from in-building location tracking systems; lighting, temperature and humidity control events from facility management systems; usage and lifecycle data management events from inventory management systems; and user events such as pulse, temperature, motion, gesture, and biometrics.

[0155] CoIs for one or more applications may include activities, behaviors, and environment condition variables such as: waypoint pattern analysis and walking speed patterns, gait, gesture, motion, non-motion, and fall patterns; smart access control card access sequence patterns; vital sign patterns; and temperature and humidity environment condition patterns.

[0156] Based on CoIs, multiple applications may insert service actions to support enhanced security and safety operations, as well as energy management and lighting control management services, using the IoT sensors and actuators. The cooperative objectives may include reduction of energy consumption, reduction of unplanned outages, and reduction of the number of safety incidents.

[0157] Embodiments may define and detect evolving CoIs including, for example, the following event patterns (denoted as C_i). CoI event C_{20} may be when a person X, enters a building, and does not pass a security check in $T_{\max}$ seconds. CoI event C_{21} may be when the person X has “walked-through” a waypoint W. Assuming that each person may have a normal work path in building, this may be expressed as {waypoint_1 -> waypoint_2, occurrence count, time intervals}. For example, CoI event C_{21} may be lobby-> second floor in 40 seconds 1.2 times a day and/or 6 times a day. It should be noted that multiple waypoint detection CoI may be defined using one or more CoIs. CoI event C_{22} may be when a security guard Y patrols check point P of building B at 2 am. CoI event C_{23} may be when a security guard Y patrols check point P of building C at 3 am. CoI event C_{24} may be the evacuation all the people from

building when fire alarm is on. CoI event C_{25} may be when a room R is vacant or occupied. CoI event C_{26} may be an IoT device usage alert (e.g., light usage hour 98%, or a light bulb is out). CoI event C_{27} may be when camera X in location L detects a customer is wandering in the store.

[0158] Service administrators of a cooperative service entity may define AEPs to detect the CoI patterns and launch a set of actions. Once the CoIs and AEPs are defined, other applications may search CoIs and insert service actions to the action list of AEPs dynamically. The following are example actions (denoted A_i) that may invoke the service function calls.

[0159] Action A_{30} may invoke an employee attendance service to track working hours. Action A_{31} may invoke a security surveillance service to validate employee identity. Action A_{32} may invoke a customer support service to help the customer. Action A_{33} may invoke a security guard management service. Action A_{34} may invoke an in-building person or motion object detection service. Action A_{35} may invoke an access control service to check if all the persons have checked out. Action A_{36} may invoke a lighting service. Action A_{37} may invoke an energy management service. Action A_{38} may invoke a building maintenance service. Action A_{39} may invoke a camera tracking and recording of a customer in location L . Action A_{40} may invoke a “dynamic service action insertion” to multiple waypoint detection points W (e.g., C_{21}).

[0160] In an embodiment, the following AEP templates (denoted AEP_i) may be created by services for A_{30} , A_{31} , A_{33} , A_{34} , and A_{36} initially. After that, new actions such as A_{31} , A_{32} , A_{34} , and A_{37} from other new services may be added to the action list incrementally. AEP template AEP_{40} may be AEP_{40} : IF C_{20} , then Do $\{A_{30}, A_{31}\}$. AEP template AEP_{41} may be AEP_{41} : IF C_{20} and C_{21} , then, Do $\{A_{31}, A_{32}\}$. AEP template AEP_{42} may be AEP_{42} : IF C_{22} and C_{23} , then, Do $\{A_{33}, A_{34}\}$. This may detect any intrusion before entering building. AEP template AEP_{43} may be AEP_{43} : IF C_{24} , then, Do $\{A_{35}, A_{34}\}$. This may verify that there are no people in the building. AEP template AEP_{44} may be AEP_{44} : IF C_{25} , then, Do $\{A_{36}, A_{37}\}$. This may control lights and air-conditioners for a room. AEP template AEP_{45} may be AEP_{45} : IF C_{26} , then, Do $\{A_{38}\}$. This may invoke

pre-emptive maintenance. AEP template AEP₄₅ may be AEP₄₅: IF C₂₇, then, Do {A₃₉, A₄₀}. On detecting wandering behavior, this may activate surveillance camera tracking and recording.

[0161] FIG. 4 is a system diagram illustrating an example service control system 400 which includes multiple IOT enabled services. In the example of FIG. 4, the physical IoT environment described above is be integrated using a virtual 3D environment, which is hosted on service server 410. Service server 410 may be implemented using a service server substantially similar to service server 190 as shown and described with respect to FIG. 1E, for example.

[0162] Service server 410 includes a tracking and detection execution environment 420, a service execution environment 430, and a shared repository 440. In some implementations, detection and execution environment 420, a service execution environment 430, and a shared repository 440 may be implemented using different servers, combined, or implemented using any other suitable topology.

[0163] Service action controllers may be implemented in the detection and execution environment 420 using a set of virtual bots, CoIBots_{1-i}. Each of CoIBots_{1-i} may be or include an autonomous controller that tracks device locations and ranges, and may map them to a virtual world. Each of CoIBots_{1-i} may track the location of a user, waypoints that the user is passing by, and/or rooms that the user enters, for example. CoIBots_{1-i} may track this information by receiving events from user devices and/or IoT sensors 450. CoIBots_{1-i} may receive such events over a suitable communications network, possibly via in intermediary such as via one or more IoT service brokers 460. Each of CoIBots_{1-i} may also track the CoIs associated with each user and may interact with multiple service functions.

[0164] Tracking and detection execution environment may identify CoIs associated with various AEPs and may transmit triggered action calls or other messages to service execution environment 430. Service execution environment 430 may receive these messages and may reply with context

aware service actions to be transmitted to users and/or IoT devices. These service actions may be provided, facilitated, and/or determined by one or more service functions 470 executing in service execution environment 430.

[0165] The service functions 470 may include such functions as virtual shopping function 471, gaming function 472, healthcare function 473, energy management function 474, or any other suitable service function 475. Service functions 470 are implemented in service execution environment 430 and offered to the user based on the CoIs for each service.

[0166] Profile information for users and devices may be stored in shared repository 440. CoIBots_{1-i} may be provided by a third party service provider which is different from the multiple service providers that provide the individual service functions in 470. Each of the service functions 470 may have a service entry point (e.g., a web service or IoT service API) and may communicate with service infrastructure 480, which may include infrastructure for providing services such as web services or IoT services. Service functions 470 may communicate with service infrastructure via a suitable communications network, possibly via an intermediary such as via one or more service brokers 490. A collection of service entry points may be grouped into a service. Each service may be monitored by a service bot (srvBot) executing on service execution environment 430. Each srvBot monitors the service action calls or messages to one or more corresponding service function. The srvBots may also interact with external service infrastructure 480 in home, office and/or cloud environments.

[0167] Operators and/or administrators of different services may create and add various service actions to the action list of selected CoIs in the repository 440 using a suitable interface, such as administration dashboard 495.

[0168] In an example scenario, a set of CoIBots may track security guards or maintenance technicians or workers. If an occurrence of event patterns that match the CoI patterns is detected, the srvBots may execute the service calls invoked by service actions from the CoIBots. Dashboard 495 may

display summaries of selected subsets of service actions, CoIs, and activities based on criteria provided by different users.

[0169] In an embodiment, CoI detection and action bots CoIBots_{1-i} may support one or more users. CoIBots_{1-i} may detect complex CoI patterns and execute service actions based on priority and IoT resource usages. CoIBots_{1-i} may monitor the effectiveness of each action based on cooperative objectives and adjust the priority of the actions based on effectiveness measurements. CoIBots_{1-i} may track the IoT usage and effectiveness measurements for quality assurance, pre-emptive maintenance, and accounting operations.

[0170] A service function processing bot (srvBot) may support dynamic subscription and registration of service action entry points, service functions, and cooperative objectives. Such srvBots may create CoIs or select CoIs created by other service providers, may insert actions to action list of AEPs of the selected CoI objects, may track the effectiveness of all the actions defined for one or more service providers, and/or may access events and tracking records stored in the existing business systems through service brokers offered by home, office, enterprise, and cloud infrastructure.

[0171] FIG. 5 is a flow chart illustrating an example method 500 for detecting and acting on CoIs. In step 510, receiving circuitry of a service server (such as service server 190, 230, 320, and/or 410 as shown and described with respect to FIGS 1E, 2, 3, or 4 respectively) receives event information from an event generating device. In step 520, updating circuitry of the service server updates context information in a memory based on the event information. Detecting circuitry of the service server detects whether the updated context information corresponds to a CoI. On a condition 530 that the updated context information corresponds to a CoI, the detecting circuitry detects whether the CoI is identified in an AEP function. Otherwise, the receiving circuitry continues to receive event information, returning to step 510. On a condition 540 that the CoI is identified in an AEP function, transmitting circuitry of the service server transmits action information to an actuating device in step 550. Otherwise, the receiving circuitry continues to

receive event information, returning to step 510.

[0172] FIG. 6 is a flow chart illustrating another example method 600 for detecting and acting on CoIs. In step 610, receiving circuitry of a service server (such as service server 190, 230, 320, and/or 410 as shown and described with respect to FIGS 1E, 2, 3, or 4 respectively) receives event information from an event generating device. In step 620, updating circuitry of the service server updates context information in a memory based on the event information. Detecting circuitry of the service server detects whether the updated context information corresponds to a CoI. On a condition 630 that the updated context information corresponds to a CoI, the detecting circuitry detects whether the CoI is identified in a first AEP function. Otherwise, the receiving circuitry continues to receive event information, returning to step 610. On a condition 640 that the CoI is identified in a first AEP function, the detecting circuitry detects whether the CoI is identified in a second AEP function. If the CoI is not identified in a second AEP function, transmitting circuitry of the service server transmits first action information to an actuating device in step 660. Otherwise, the detecting circuitry detects whether an action of the second AEP function conflicts with the first action information. On a condition 670 that the action of the second AEP does not conflict with the first action information, the transmitting circuitry transmits second action information in a step 680. Otherwise, if the action of the second AEP does conflict with the first action information, the first and/or second action information is modified or omitted to resolve the conflict, and the first and second action information are transmitted to one or more actuating devices in steps 680 and 660.

[0173] FIG. 7 is a flow chart illustrating an example method 700 for configuring and employing AEPs. In step 710, receiving circuitry of a service server (such as service server 190, 230, 320, and/or 410 as shown and described with respect to FIGS 1E, 2, 3, or 4 respectively) receives an AEP and a list of actions corresponding to the AEP, each of the actions in the list of actions having a corresponding priority. In step 720, the receiving circuitry

receives event information from an event generating device. Detecting circuitry of the service server detects whether the event information satisfies the AEP. On a condition 730 that the event information satisfies the AEP, the selecting circuitry of the service server selects an action from the list based on priority in step 740, and transmitting circuitry of the service server transmits action information based on the selected action to an actuating device in step 750. Otherwise, if the event does not satisfy the AEP, the flow returns to step 720.

[0174] In some implementations, an effectiveness of the action may be measured and the priority of one or more of the list of actions may be modified based on the effectiveness. The effectiveness may be measured, for example, in terms of cost or return-on-investment of the action, for example, or any other suitable measurement. In some implementations, determining circuitry of the service server determines whether the highest priority action conflicts with another action (e.g., reduces the efficacy of another action already in progress based on another AEP) and if so, selects the next lower priority action, determining whether that action conflicts with another action, and so forth.

[0175] The embodiments described above provide a new cooperative service control system that may allow for multiple independently developed service functions to cooperate and offer service actions to users and devices dynamically, incrementally, and efficiently. The methods, devices, and systems may be deployed to distributed IoT environments with standard communication protocols and service brokers supporting event subscription and publication APIs. A set of methods and service interfaces provided by the proposed system may support dynamic creation of contexts of interests based on combinations of real-time event patterns, user behavior metrics, activities, and environment conditions. The CoI creation, detection and service action control executions may be decoupled from the service function creation and execution. Each application and service may insert new service actions to the action list to invoke remote execution of service functions dynamically. The

decoupled detection, action control, and service functions execution may reduce the duplicated design and processing efforts. In addition, coordination of actions at the moments of engagement defined by shared CoIs may create opportunities for multiple service providers to offer bundled services cooperatively and incrementally while minimizing resource conflicts and service disruption. Effectiveness measurements of cooperative objectives may support incremental evolutions and improvements of service actions and operation efficiency offered by one or more service providers.

[0176] Although features and elements are described above in particular combinations, one of ordinary skill in the art will appreciate that each feature or element can be used alone or in any combination with the other features and elements. In addition, the methods described herein may be implemented in a computer program, software, or firmware incorporated in a computer-readable medium for execution by a computer or processor. Examples of computer-readable media include electronic signals (transmitted over wired or wireless connections) and computer-readable storage media. Examples of computer-readable storage media include, but are not limited to, a read only memory (ROM), a random access memory (RAM), a register, cache memory, semiconductor memory devices, magnetic media such as internal hard disks and removable disks, magneto-optical media, and optical media such as CD-ROM disks, and digital versatile disks (DVDs). A processor in association with software may be used to implement a radio frequency transceiver for use in a WTRU, UE, terminal, base station, RNC, or any host computer.

*

*

*

CLAIMS

What is claimed is:

1. A method for controlling context aware internet-of-things (IoT) services, the method comprising:
 - receiving, by receiving circuitry of a service server, an actionable event pattern (AEP) and a list of actions corresponding to the AEP, wherein each action in the list of actions has a priority;
 - receiving, by the receiving circuitry of the service server, event information from an event generating device;
 - detecting, by detecting circuitry of the service server, whether the event information satisfies the AEP;
 - on a condition that the event information satisfies the AEP, selecting an action from the list of actions based on the priority of the actions in the list of actions, and transmitting, by transmitting circuitry of the service server, action information based on the selected action, to an actuating device.
2. The method of claim 1, further comprising:
 - receiving, by the receiving circuitry of the service server, feedback regarding an action performed by the actuating device based on the action information transmitted to the actuating device; and
 - measuring, by measuring circuitry of the service server, an effectiveness of the action.
3. The method of claim 2, further comprising:
 - adjusting, by adjusting circuitry of the service server, the priority of the action based on the measured effectiveness.

4. The method of claim 2, further comprising:
adjusting, by adjusting circuitry of the service server, the priority of at least one action of the list of actions based on the measured effectiveness.
5. The method of claim 2, wherein the effectiveness comprises a cost associated with the action.
6. The method of claim 2, wherein the feedback is received by the service server from the actuating device.
7. The method of claim 1, wherein the event information comprises information regarding a user behavior.
8. The method of claim 1, wherein the event information comprises IoT sensor data.
9. The method of claim 1, wherein selecting the action from the list of actions based on the priority of the actions in the list of actions comprises:
selecting a highest priority action from the list of actions which does not conflict with another action.
10. The method of claim 9, wherein the action does not conflict with another action on a condition that the action does not reduce an effectiveness of the other action.
11. A service server for controlling context aware internet-of-things (IoT) services, the service server comprising:
a processor and a non-transitory computer readable memory;

receiving circuitry for receiving an actionable event pattern (AEP) and a list of actions corresponding to the AEP, wherein each action in the list of

actions has a priority, and for receiving, event information from an event generating device;

detecting circuitry for detecting whether the event information satisfies the AEP;

selecting circuitry for, on a condition that the event information satisfies the AEP, selecting an action from the list of actions based on the priority of the actions in the list of actions; and

transmitting circuitry for, on a condition that the event information satisfies the AEP, transmitting action information based on the selected action, to an actuating device.

12. The service server of claim 11, wherein:

the receiving circuitry of the service server is further configured to receive feedback regarding an action performed by the actuating device based on the action information transmitted to the actuating device; and

wherein the service server further comprises measuring circuitry for measuring an effectiveness of the action.

13. The service server of claim 12, further comprising:

adjusting circuitry of the service server for adjusting the priority of the action based on the measured effectiveness.

14. The service server of claim 12, further comprising:

adjusting circuitry of the service server for adjusting priority of at least one action of the list of actions based on the measured effectiveness.

15. The service server of claim 12, wherein the effectiveness comprises a cost associated with the action.

16. The service server of claim 12, wherein the feedback is received by the service server from the actuating device.

17. The service server of claim 11, wherein the event information comprises information regarding a user behavior.

18. The service server of claim 11, wherein the event information comprises IoT sensor data.

19. The service server of claim 11, wherein selecting the action from the list of actions based on the priority of the actions in the list of actions comprises:

selecting the action from the list of actions having a highest priority which does not conflict with another action.

20. The service server of claim 19, wherein the action does not conflict with another action on a condition that the action does not reduce an effectiveness of the other action.

21. A method for controlling context aware internet-of-things (IoT) services, the method comprising:

receiving, by receiving circuitry of a service server, event information from an event generating device;

updating, by updating circuitry of the service server, context information in a memory based on the event information;

detecting, by detecting circuitry of the service server, whether the updated context information corresponds to a context of interest (CoI);

on a condition that the updated context information corresponds to a CoI, detecting, by the detecting circuitry, whether the CoI is identified in an actionable event pattern (AEP) function;

on a condition that the CoI is identified in an AEP function; transmitting, by transmitting circuitry of the service server, action information to an actuating device.

22. The method of claim 21, further comprising transmitting, by the transmitting circuitry of the service server, a second action information to a second actuating device on a condition that a second AEP identifies the same CoI.

23. The method of claim 21, further comprising modifying the action information on a condition that a second AEP identifies the same CoI.

24. The method of claim 21, further comprising refraining from transmitting the action information on a condition that a second AEP identifies the same CoI.

25. The method of claim 21, further comprising receiving, by the receiving circuitry of the service server, configuration information for the CoI from a service provider via a configuration interface.

26. The method of claim 21, further comprising receiving, by the receiving circuitry of the service server, configuration information for the AEP from a service provider via a configuration interface.

27. The method of claim 21, further comprising receiving, by the receiving circuitry of the service server, configuration information for the CoI from a service provider via a configuration interface, and receiving second configuration information for a second CoI from a second service provider via the configuration interface.

28. The method of claim 21, further comprising receiving, by the receiving circuitry of the service server, configuration information for the AEP from a service provider via a configuration interface, and receiving second

configuration information for a second AEP from a second service provider via the configuration interface.

29. The method of claim 21, wherein the event generating device comprises an IoT sensor.

30. The method of claim 21, wherein the actuating device comprises an IoT actuator.

31. A service server for controlling context aware internet-of-things (IoT) services, the service server comprising:

- a processor and a non-transitory computer readable memory;
- receiving circuitry for receiving event information from an event generating device;
- updating circuitry for updating context information in the memory based on the event information;
- detecting circuitry for detecting whether the updated context information corresponds to a context of interest (CoI), and on a condition that the updated context information corresponds to a CoI, detecting whether the CoI is identified in an actionable event pattern (AEP) function; and
- transmitting circuitry for transmitting action information to an actuating device on a condition that the CoI is identified in an AEP function.

32. The service server of claim 31, wherein the transmitting circuitry transmits a second action information to a second actuating device on a condition that a second AEP identifies the same CoI.

33. The service server of claim 31, further comprising modifying circuitry for modifying the action information on a condition that a second AEP identifies the same CoI.

34. The service server of claim 31, wherein the transmitting circuitry refrains from transmitting the action information on a condition that a second AEP identifies the same CoI.

35. The service server of claim 31, wherein the receiving circuitry comprises a configuration interface for receiving configuration information for the CoI from a service provider.

36. The service server of claim 31, wherein the receiving circuitry comprises a configuration interface for receiving configuration information for the AEP from a service provider.

37. The service server of claim 31, wherein the receiving circuitry comprises a configuration interface for receiving configuration information for the CoI from a service provider via a configuration interface, and for receiving second configuration information for a second CoI from a second service provider.

38. The service server of claim 31, wherein the receiving circuitry comprises a configuration interface for receiving configuration information for the AEP from a service provider via a configuration interface, and for receiving second configuration information for a second AEP from a second service provider.

39. The service server of claim 31, wherein the event generating device comprises an IoT sensor.

40. The service server of claim 31, wherein the actuating device comprises an IoT actuator.

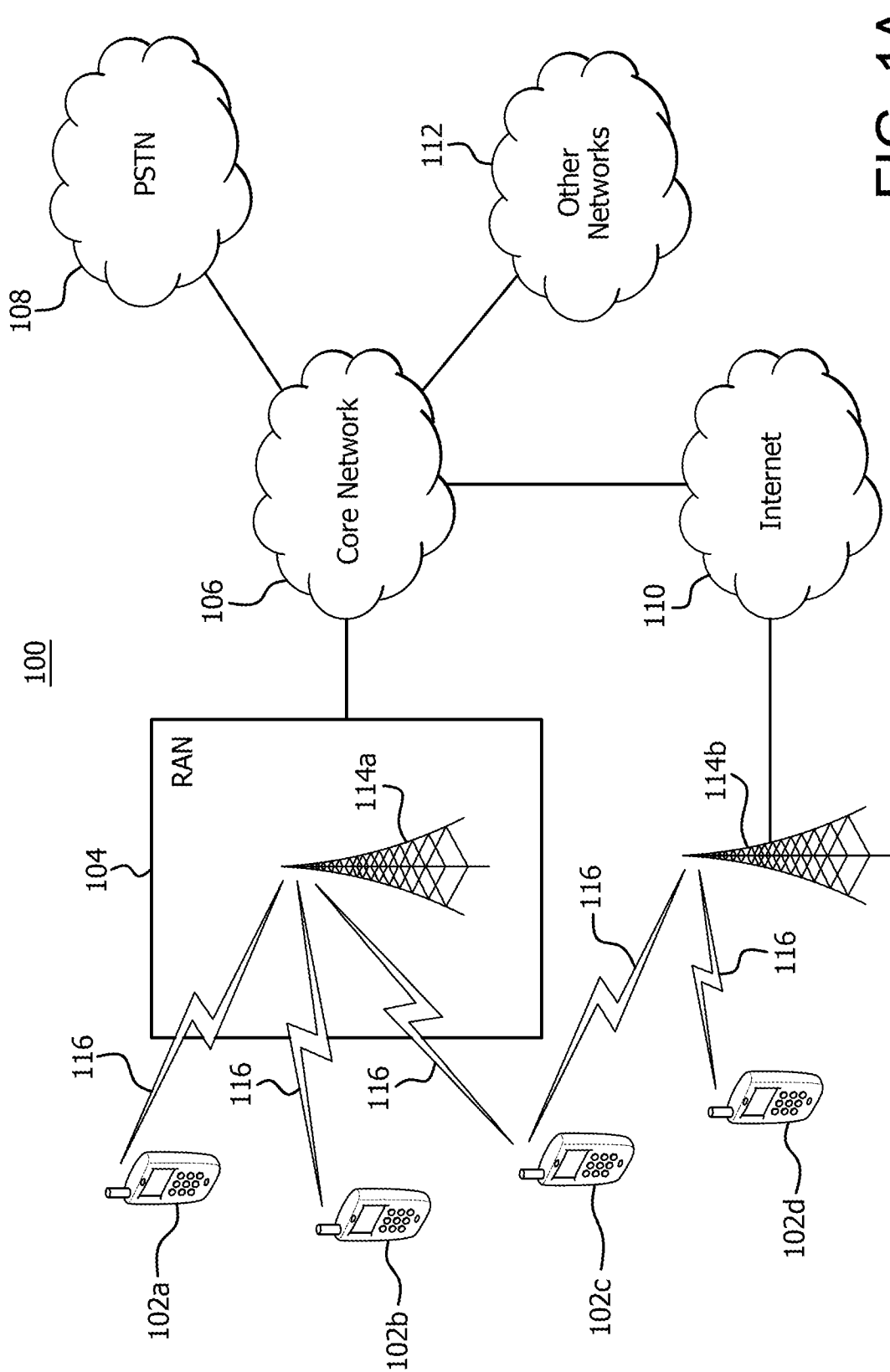


FIG. 1A

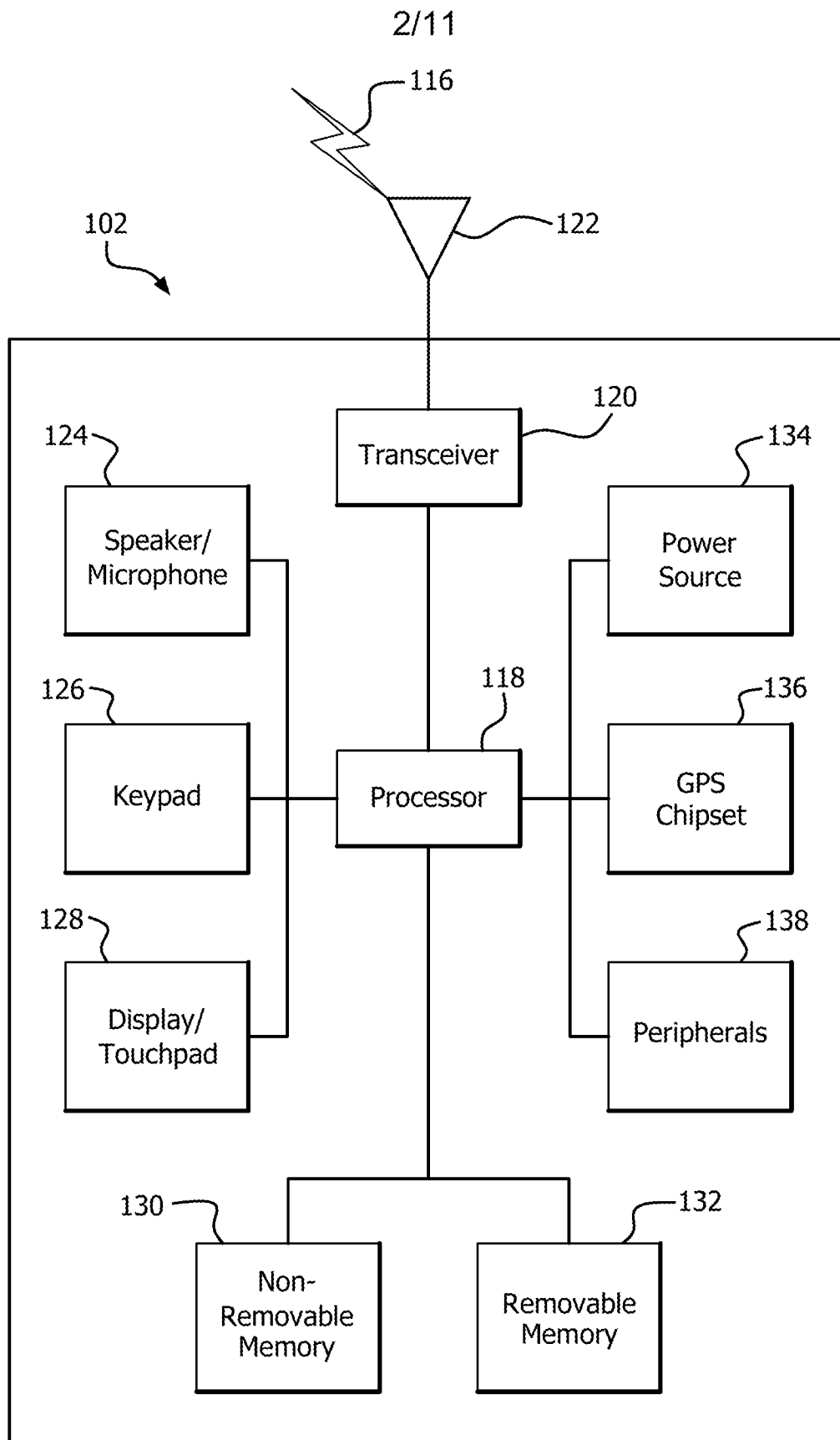


FIG. 1B

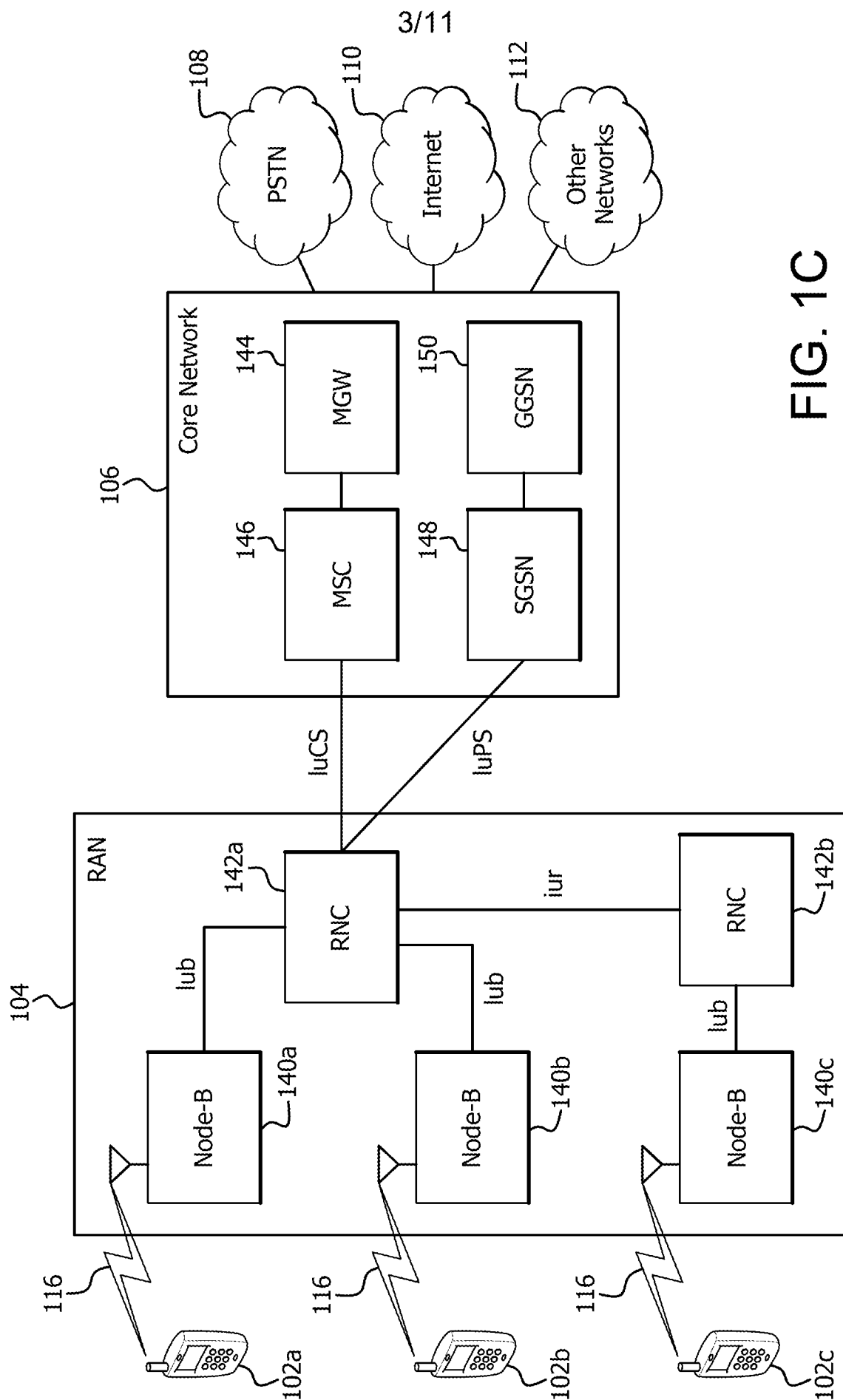


FIG. 1C

4/11

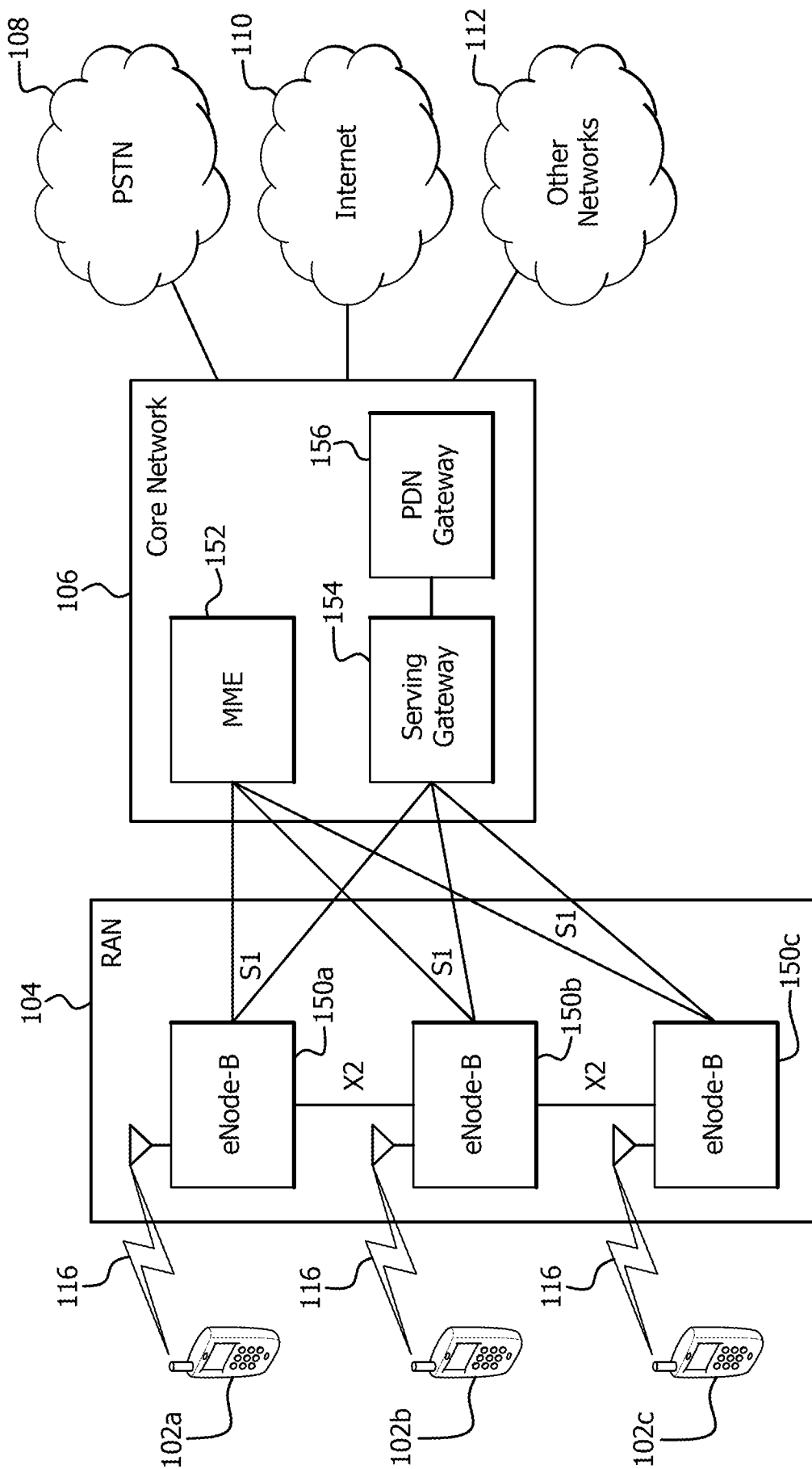


FIG. 1D

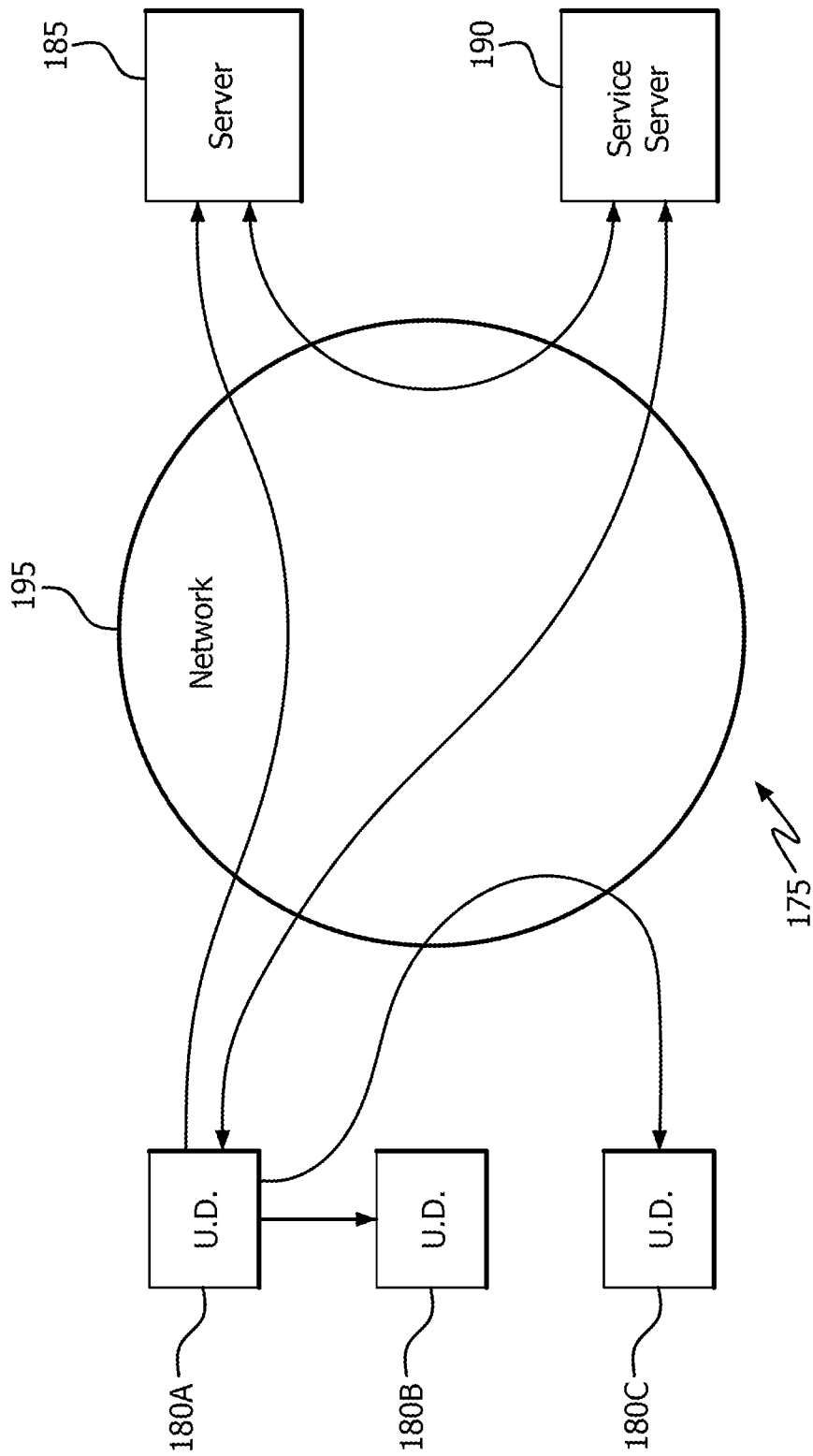


FIG. 1E

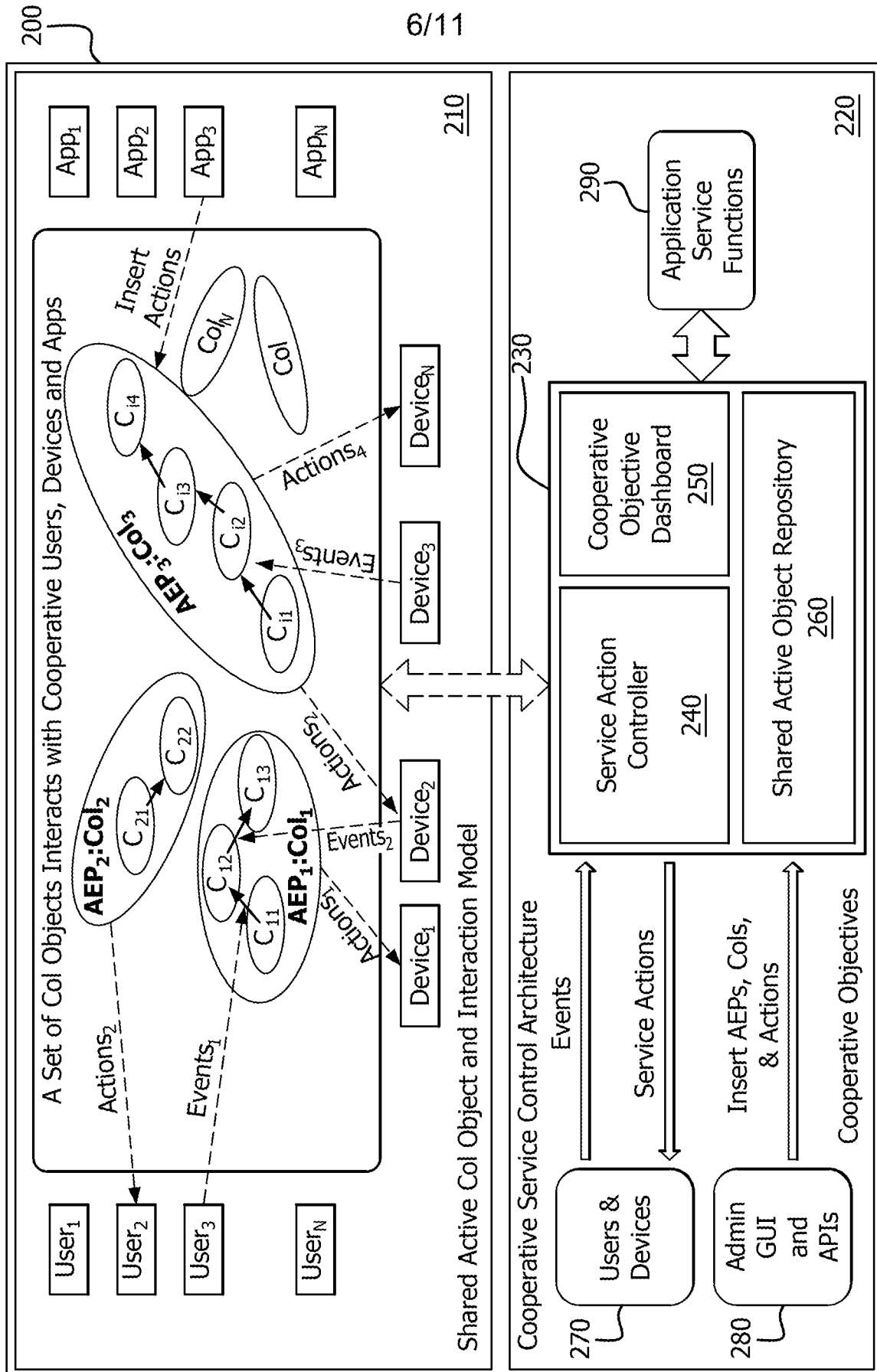


FIG. 2

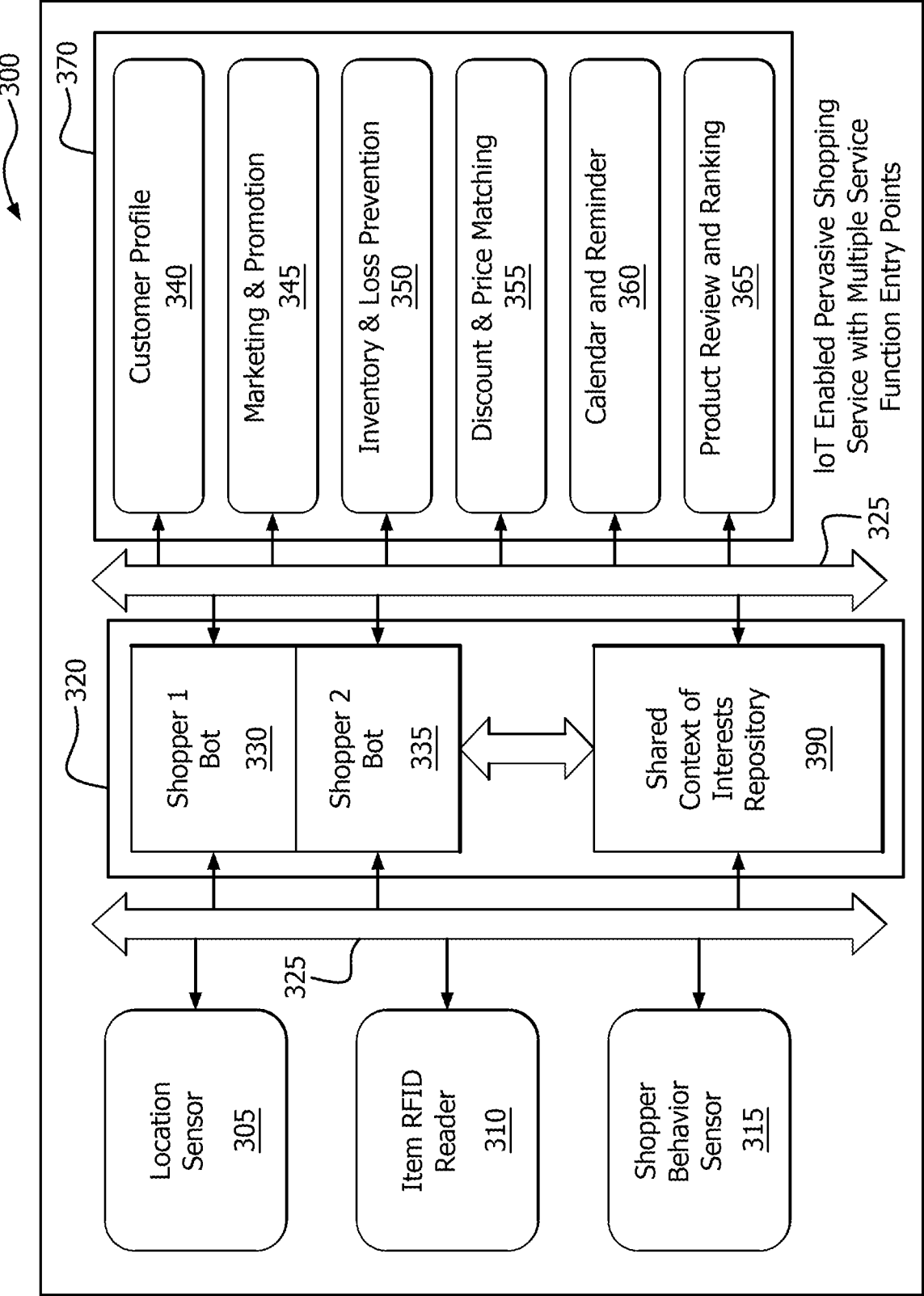


FIG. 3

8/11

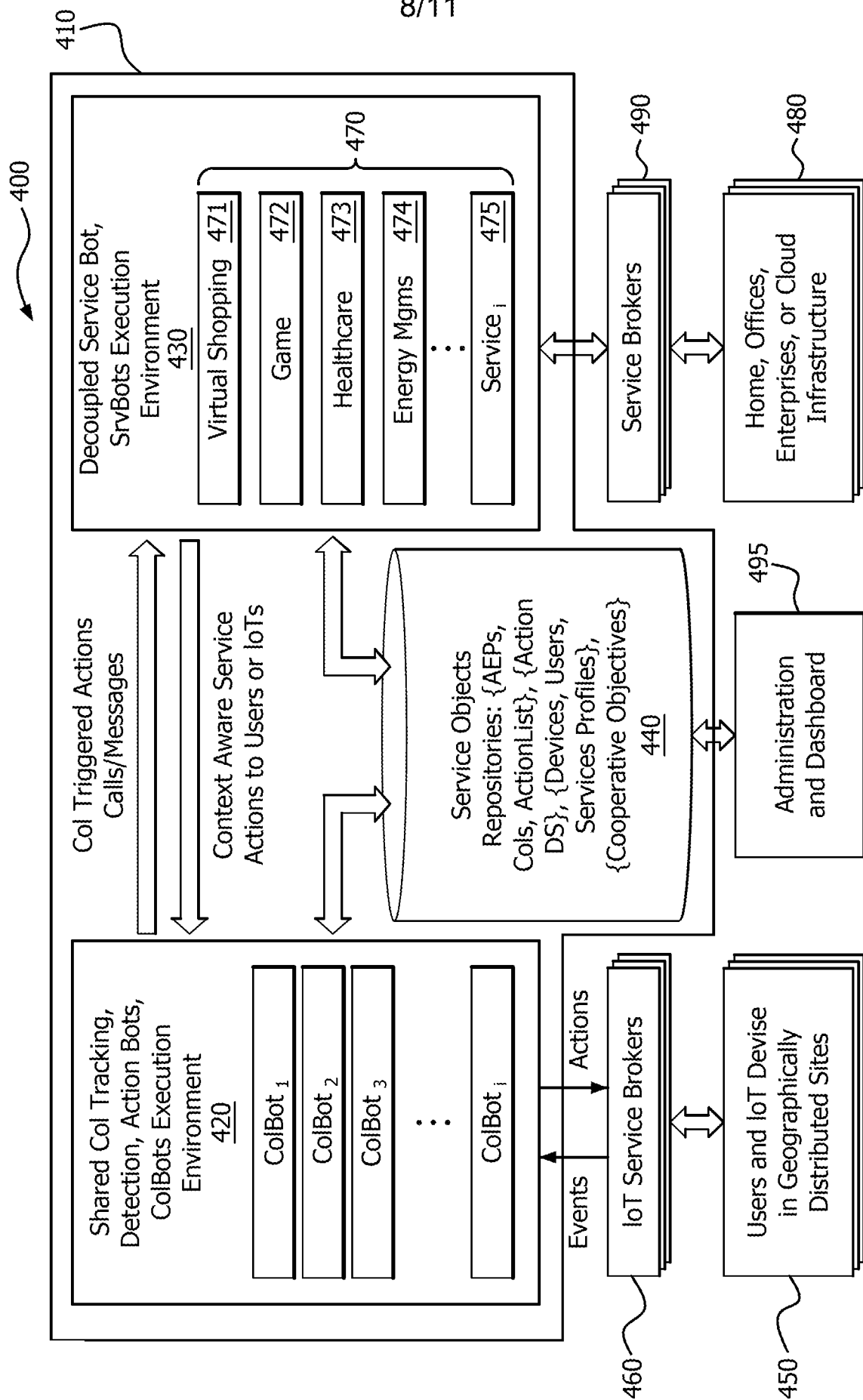


FIG. 4

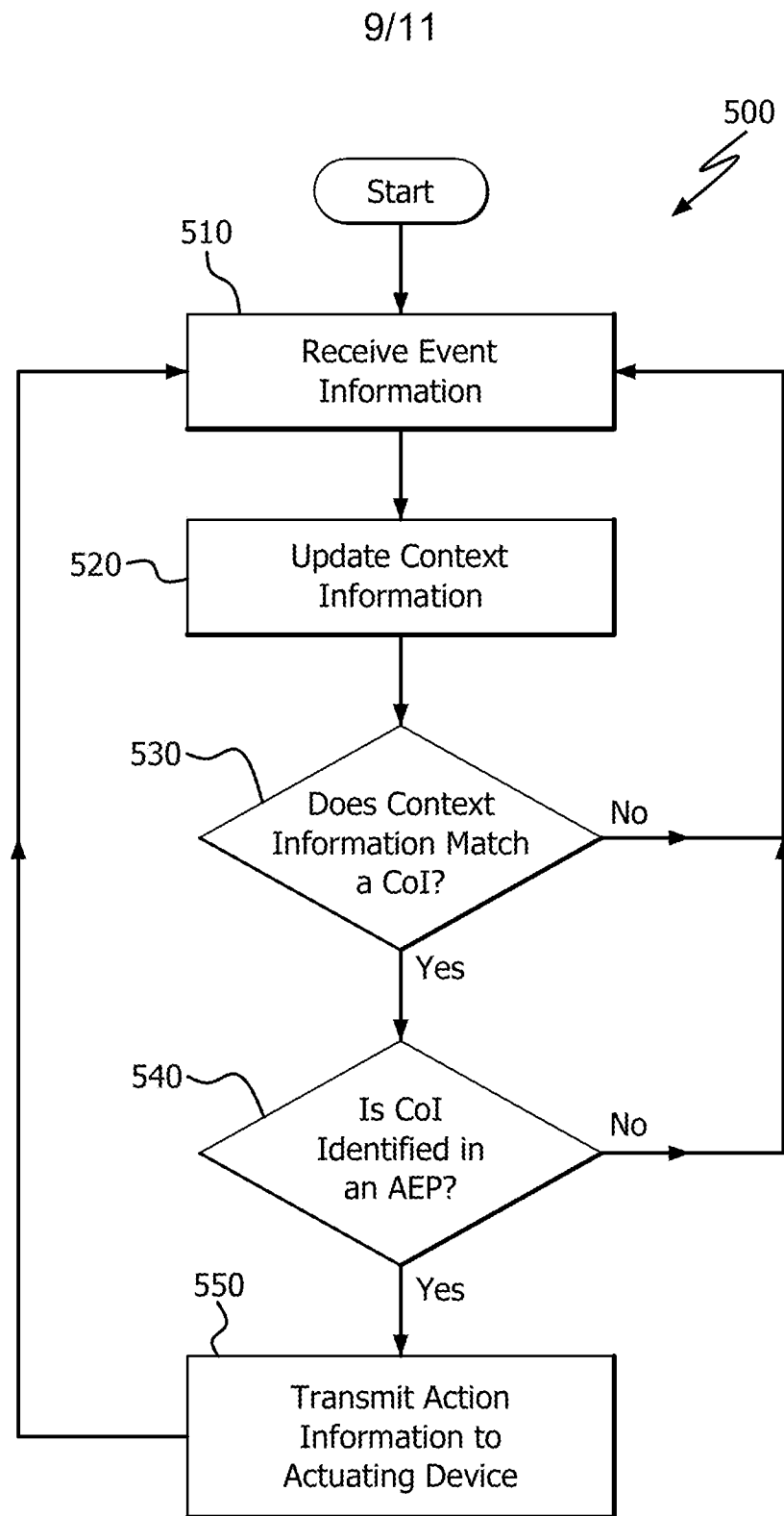


FIG. 5

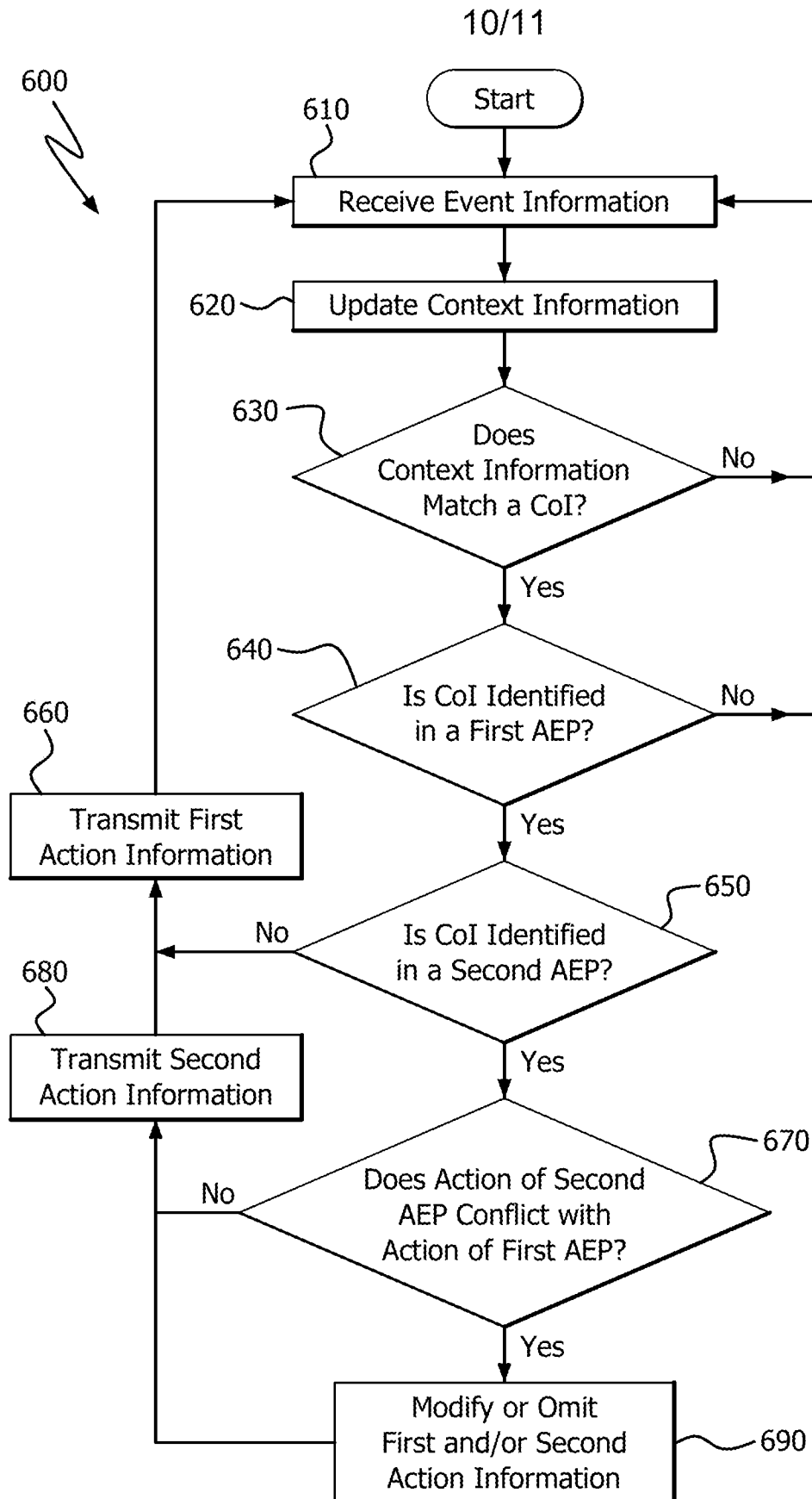


FIG. 6

11/11

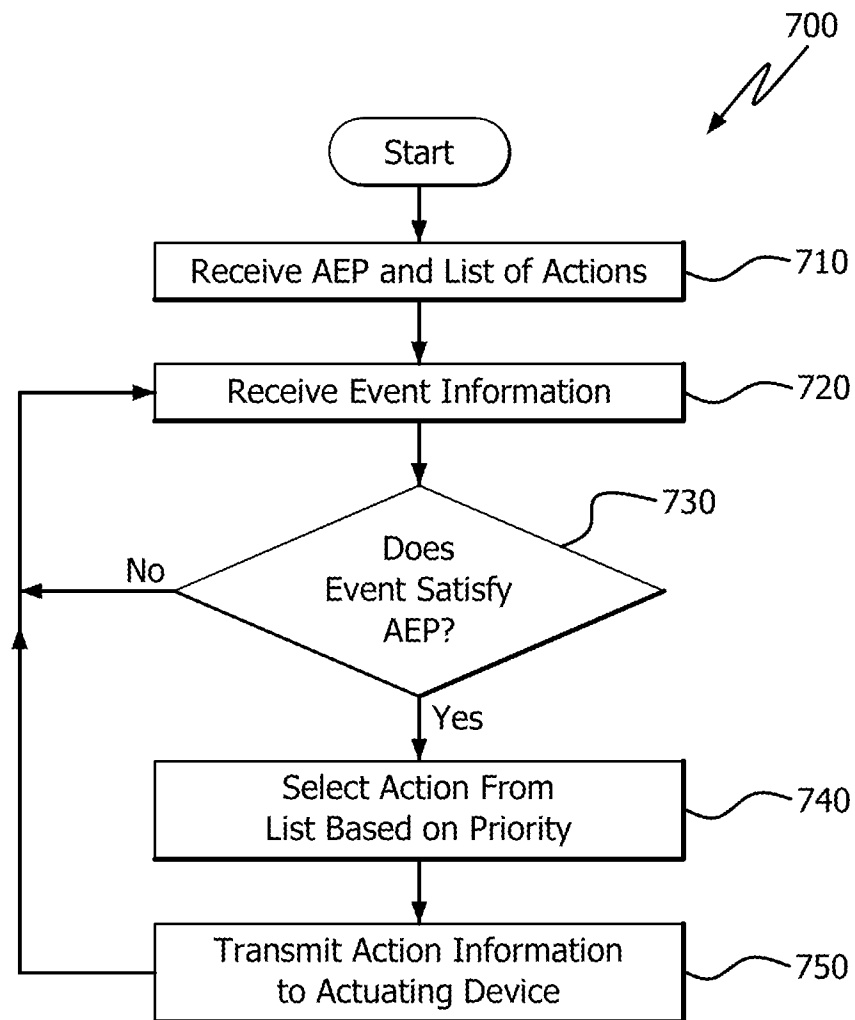


FIG. 7

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2016/037623

A. CLASSIFICATION OF SUBJECT MATTER

INV. H04W4/00 H04L29/08
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04W H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, IBM-TDB, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2015/019714 A1 (SHAASHUA TRIINU MAGI [US] ET AL) 15 January 2015 (2015-01-15) paragraph [0066] - paragraph [0084] -----	1-40
A	US 2015/089576 A1 (POROOR JAYARAJ [IN] ET AL) 26 March 2015 (2015-03-26) paragraph [0037] - paragraph [0045] -----	1-40
A	US 2014/244710 A1 (SHARMA SANDEEP [US] ET AL) 28 August 2014 (2014-08-28) abstract -----	1-40



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

31 August 2016

Date of mailing of the international search report

07/09/2016

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Suciu, Radu

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2016/037623

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2015019714 A1	15-01-2015	EP 3019919 A1	18-05-2016
		US 2015019714 A1	15-01-2015
		WO 2015006080 A1	15-01-2015

US 2015089576 A1	26-03-2015	NONE	

US 2014244710 A1	28-08-2014	CN 105074684 A	18-11-2015
		EP 2959396 A2	30-12-2015
		JP 2016517638 A	16-06-2016
		KR 20150121113 A	28-10-2015
		TW 201437944 A	01-10-2014
		US 2014244710 A1	28-08-2014
		WO 2014131029 A2	28-08-2014
